



# AUTOMATISK KALIBRERING AF RUNDKØRSLER I VISSIM VED HJÆLP AF GENETISK ALGORITME

KANDIDATSPECIALE – ANDERS YDE-NIELSEN – VEJE & TRAFIK

## TITELBLAD

**Titel:** Automatisk kalibrering af rundkørsler i VISSIM ved hjælp af genetisk algoritme.

**Projekttype:** Kandidatspeciale

**Nøgleord:**

**Projektperiode:** 1. februar 2018 – 08 juni 2018.

**Omfang:** 30 ETCS

**Forfatter:** Anders Yde-Nielsen

**Vejleder:**

Harry Lahrmann, Aalborg Universitet

**Censor:**

Steen Merlach Lauritzen, Vejdirektoratet

**Oplagstal:** 3

**Sidetæl:** 72

## SYNOPSIS

Dette kandidatspeciale omhandler mulighederne for at skabe et system, som kan kalibrere mikrosimuleringsmodeller på baggrund af virkelig data indsamlet med COWIs DataFromSky og generelle betragtninger i forhold til modellering af virkeligheden.

Ved hjælp af MATLAB og VISSIMs COM-module er der skabt et system, som kan ændre på udvalgte kalibreringsparametre fra VISSIM. Med udgangspunkt i litteraturstudiet og faglige samtaler med COWI er der udvalgt fem kalibreringsparametre, som løbende ændres for at finde det optimale parametersæt til VISSIM i forhold til rejsetider trukket ud af DataFromSky. Der er arbejdet med flere optimeringsalgoritmer, hvor de mere avancerede algoritmer giver bedste og mest konsistente resultater.

På trods af en vellykket kalibrering, viser resultaterne, at VISSIM har problemer med at modellere rejsetiderne præcist ved trængsel. Ved relativt lave belastninger har VISSIM ingen problemer.



**AALBORG UNIVERSITET**  
STUDENTERRAPPORT

Institut for Byggeri & Anlæg  
Thomas Manns Vej 23  
9220 Aalborg Ø  
Telefon: 99 40 84 84  
E-mail: [civil@civil.aau.dk](mailto:civil@civil.aau.dk)  
[www.byggeri.aau.dk](http://www.byggeri.aau.dk)

i samarbejde med:



Parallelvej 2  
2800 Lyngby  
Telefon: 56 40 00 00  
E-mail: [cowi@cowi.dk](mailto:cowi@cowi.dk)  
[www.cowi.com](http://www.cowi.com)

## ABSTRACT

This master thesis investigates a possible set up of automatic calibration of microsimulation models. The automatic calibration is based on matching observed travelling times from a project location obtained by COWI's DataFromSky, with travelling times from the same location modelled by microsimulation in VISSIM. This is done by variation of five chosen parameters within driving behaviors and conflict areas. The five variables together comprise a set.

Two methods are investigated. One method generates a specified number of sets with random parameters and evaluates the difference in travelling times. Another method chooses the variables in the sets based on the difference in travelling times from previous simulations, which allows for an adaptive change of the parameters. The parameters are changed in the model by the help of MATLAB and VISSIM's COM-module.

The calibration process that involves the genetic algorithm has been proven to be

successful in terms of yielding improved results through the generations. However, the adaptive change of the chosen parameters has not been sufficient to yield a model that represents the actual behavior of the traffic during congestion in the most loaded hours of the day.



## FORORD

Denne rapport er et kandidatspeciale, som er udarbejdet af undertegnede på kandidatuddannelsen *Trafik og Veje* ved Aalborg Universitet Institut for Byggeri & Anlæg i perioden 1. februar 2018 til 8. juni 2018.

Projektet udspringer af en nysgerrighed og et ønske om at optimere kalibreringsprocessen i mikrosimuleringsprogrammet VISSIM og skal ses om et alternativ til den normale manuelle kalibrering, som foretages af modelbyggeren. Formålet med projektet er derfor, at opbygge et system, som kan evaluere på VISSIM simuleringer i forhold til virkelig data og herefter selv foretage kalibreringsprocessen. På den måde vil kalibreringen blive mere præcis og samtidig vil ingeniøren som normalt ville kalibrere modellen have tid til andet arbejdet.

Ydermere vil VISSIM evne til at modellere virkeligheden blive undersøgt, hvor der benyttes virkelige data til at sammenligne modellens resultater.

I forbindelse med projektet har jeg anvendt COWIs DataFromSky til indhente virkelige data, som danner baggrunden for disse kalibreringer. Derfor vil jeg gerne takke COWI og ikke mindst projektchef Jonas Hammershøj Olesen og sektionsleder Louise Lundbeck Krog, som har gjort det muligt at benytte denne software.

Der rettes en tak til følgende personer, som har hjulpet med konstruktiv dialog, databehandling og indsamling:

Line Degn Andersen, COWI  
Mikkel Færgemand, COWI  
Rasmus Albrink, COWI

Derudover vil jeg gerne rette en særlig tak Andreas Berre Eriksen, Ph.d.-studerende ved Aalborg Universitet. Andreas har hjulpet med det programtekniske og opsætningen af de i projektet anvendte optimeringsalgoritmer. Samtidig har han været garant for sparring i forhold til valg og metode til optimeringsalgoritmerne, som ellers ligger udenfor vores normale fagområde.

En stor tak skal også rettes til lektor og vejleder Harry Lahrmann, som har bidraget med konstruktiv kritik, inspirerende samtaler og vejledning.

---

Anders Yde-Nielsen

## LÆSEVEJLEDNING

Denne projektrapport udgør den skriftlige del af kandidatspecialet, og har til formål at dokumentere metode, resultater i forbindelse med den automatiserede kalibrering af VISSIM modeller. Da projektet er præget af mikrosimulering samt en del kodning og optimeringsanalyser vil visse afsnit fremstå tekniske og til tider kompliceret. Rapporten henviser sig derfor til fagfolk, som har kendskab til enten VISSIM eller optimeringsprocesser, og derved ikke så meget til folk uden kendskab til førnævnte områder. Der er i rapporten anvendt tekniske begreber, som bliver forklaret i takt med at de bliver anvendt. Dertil er der løbende i projektet anvendt illustrationer for at kunne forklare tekniske begreber lettere. Indledningsvis er der udarbejdet en ordliste, som læseren kan støtte sig op ad, hvor nogle af de mere tekniske begreber er forklaret kort.

I starten af hvert kapitel er der med kursiv angivet en beskrivelse af, hvad det indeværende kapitel vil indeholde, så læseren hurtigt får et overblik over hvad

de kan forvente. For større afsnit er dette også anvendt.

I rapporten er angivet referencer, som alle er samlet i alfabetisk rækkefølge i litteraturlisten bagerst i rapporten. Til disse referencer benyttes Harvardmetoden, som i teksten er angivet (forfatter, år).

Tabeller og figurer er nummeret i henhold til det enkelte kapitel eksempelvis figur 1-1 efterfulgt af figur 1-2 i kapitel 1. Hvis der ikke fremgår referencer på figurerne er de udarbejdet af undertegnede.

I rapporten vises et udsnit af det projektarbejde, som er foregået, hvor der er præsenteret et repræsentativt udsnit af resultaterne og hovedpointerne er taget op. De mere detaljerede resultater med henvisning til bilag bagerst i rapporten. Resultaterne i rapporten afrundet for ikke at forvirre læseren med unødvendigt detaljerede resultater. Alle decimaler er brugt i udregningerne.

Udover bilaget som er bagerst i rapporten er der vedlagt et elektronisk bilag indeholdende alle MATLAB scripts.

## ORDFORKLARING

### **Brute Force Algoritme**

Algoritme som danner tilfældige parametersæt og kører dem igennem uden nogen løbende optimering.

### **Child**

Children bliver genereret på baggrund af Parents ved hjælp af Elitechild, Mutation eller Crossover.

### **Conflict Area**

Metode til modellering af vigepligt i VISSIM. Denne metode er den anbefalede af PVT Group, som er skaberen af VISSIM.

### **Crossover**

Proces i GA, som krydser to parents værdier og dermed skaber nye children, kaldt CrossoverChildren.

### **DataFromSky**

Databehandlingsprogram udviklet af RCE Systems, som COWI har monopol på i norden. Kan behandle trackede drone optagelser og benyttes til at trække data ud fra dem.

### **Driving Behavior**

Metode til at modellere kørselsadfærd i VISSIM, som er baseret på Wiedemanns forfølgelsesmodel fra 1974.

### **Elitechild**

Proces i GA, som udvælger de bedste parents og viderefører dem direkte til næste generation uden mutation eller crossover.

### **Gates (entry- & exitgates)**

Et snit i DataFromSky, som opsamler data fra trajectories og logger køretøjer som passerer snittet. Data såsom køretøjs nummer, køretøjstype, hastighed, acceleration mm. bliver logget for hver trafikant.

### **Generation**

GA algoritmen er baseret på generationer, hvor hver generation har en vis population. Individerne i populationen ændres i takt med generationerne.

### **Genetisk Algoritme (GA)**

Optimerings algoritme baseret på Darwins Evolutions teori, som ved hjælp af generationer kommer tættere og tættere på et resultat.

### **Individual**

Et individ (i dette tilfælde et parametersæt) som benyttes i genetiske algoritmer.

### **Kalibreringsmål**

Se "Score".

### **Mutation**

Proces i GA, som muterer parents med en vis sandsynlighed, og dermed skaber nye children, kaldt MutationChildren.

### **OD Matrice**

Origin-destination matrice, som beskriver, hvor mange trafikanter, der benytter de forskellige ruter.

### **Parent**

Et udvalgt individual i en generation, som danner baggrund for individerne (children) i næste generation.

### **Population**

Antallet af individer udgør populationen.

### **Random Seed**

Et specifikt nummer som beskriver hvilken tilfældighed en given VISSIM simulering er foretaget med. De forskellige

Random Seeds sørger for at ankomstfordelingen, og andre tilfældige faktorer, bliver anderledes for hver simulering.

**Reduced Speed Area**

Metode til at modellere lokale hastighedsnedsættelser i sving og lignende i VISSIM.

**Score**

Mål brugt i alle optimeringsalgoritmer til at bestemme, hvor langt de simulerede modeller er fra virkeligheden. I denne rapport er forskellen i rejsetid brugt som score. En god score er en lav score.

**Selection**

Proces i GA, som udvælger parents, der skal bruges i næste generation. Denne Selection er baseret på individernes score, således der er større chance for at gode individer kommer med i næste generation.

**Trajectory**

Der bliver dannet en trajectory for hver trafikant, som viser trafikantens vej igennem optagelsen.

---

## INDHOLDSFORTEGNELSE

|                                              |           |                                            |           |
|----------------------------------------------|-----------|--------------------------------------------|-----------|
| <b>1. Baggrund</b>                           | <b>1</b>  | <b>4. Diskussion</b>                       | <b>49</b> |
| 1.1 Trafikmodellering                        | 2         | 4.1 Forudsætninger/metode                  | 49        |
| 1.2 Mikrosimulering                          | 4         | 4.2 Resultater                             | 51        |
| 1.3 Maskinlæring og optimering               | 9         | 4.3 Fremtidig arbejde                      | 53        |
| 1.4 Opsamling af litteraturstudie            | 12        | <b>5. Konklusion</b>                       | <b>55</b> |
| 1.5 Projektlokalitet                         | 14        | <b>6. Litteraturliste</b>                  | <b>57</b> |
| 1.6 Projektbeskrivelse                       | 14        | <b>A. Litteraturstudie</b>                 | <b>59</b> |
| 1.7 Projektafgrænsning                       | 15        | <b>B. Rejsetider fra DFS</b>               | <b>63</b> |
| <b>2. Metode</b>                             | <b>17</b> | <b>C. Resultater fra VISSIM analyse</b>    | <b>65</b> |
| 2.1 Litteratursøgning                        | 17        | <b>D. Resultater fra 30 minuttersmodel</b> | <b>69</b> |
| 2.2 Dataindsamling                           | 18        |                                            |           |
| 2.3 Databehandling                           | 22        |                                            |           |
| 2.4 Trafikmodellering                        | 24        |                                            |           |
| 3.2 Automatiseret kalibrering                | 27        |                                            |           |
| <b>3. Resultater</b>                         | <b>35</b> |                                            |           |
| 3.1 Opbygning af VISSIM Model                | 35        |                                            |           |
| 3.2 Optimeringsalgoritmer                    | 39        |                                            |           |
| 3.3 Sammenligning af det bedste parametersæt | 45        |                                            |           |
| 3.4 30 minuttersmodel                        | 47        |                                            |           |

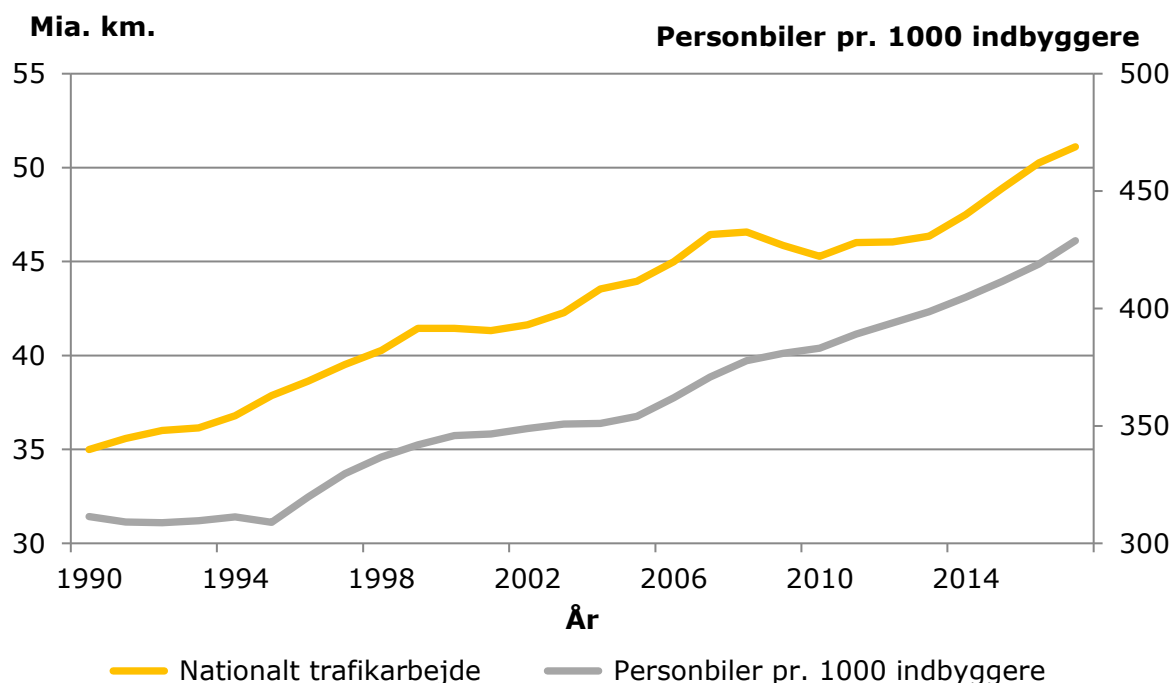




## 1 BAGGRUND

De seneste årtier har der i Danmark været en markant stigning i biltrafikken og på nuværende tidspunkt er der ikke noget, som tyder på, at der vil komme en reduktion i trafikarbejdet i den nærmeste fremtid. I de seneste fem år er trafikken steget med over 8%, som det fremgår af figur 1-1, og i 2016 forventes trafikarbejdet at runde 50 milliarder kilometer årligt. De største stigninger findes på de danske motorveje, og selvom motorvejene udgør mindre end 2% af den samlede længde af vejnettet, blev 33% af vejtrafikken afviklet på motorvejene i 2016. På flere af de mest trafikkede motorveje har der været stigninger helt op til 9% fra 2015 til 2016 (Vejdirektoratet, 2018). Ifølge Vejdirektoratet kan der forventes en stigning på ca. 8% fra 2015-2020, hvilket svarer til en årlig stigning på 1,5%. (Vejdirektoratet, 2018)

Trafikvæksten hænger sammen med at flere danskere har fået rådighed over en



**Figur 1-1 Stigning i det nationale trafikarbejde fra 1950 til 2015.**

bil. Antallet af nyregistrerede personbiler slog i 2016 rekord for sjette år i træk. Der blev i 2016 nyregistreret 222.000 personbiler, hvilket er en vækst på knap 11% i forhold til 2015. I 2016 har 60% af danske familier mindst én bil. Det bør ses i lyset af, at der de senere år er sket en ændring i, hvor langt folk er villige til at pendle til og fra arbejde. Danskerne er villige til at bosætte sig længere væk

fra deres arbejdsplads end tidligere, hvilket åbner op for mere pendling end tidligere. Det er derfor nødvendigt for flere familier at have en bil, og for visse familier to biler, da der skal pendles op mod to timer i løbet af dagen. (Vejdirektoratet, 2018)

Stigningerne i både trafikarbejdet og bilparken skyldes blandt andet den højreorienterede regerings fokus på at sænke registreringsafgifterne, hvilket gør det billigere for den enkelte dansker at købe nye biler. Specielt Liberal Alliance har arbejdet målrettet mod at få registreringsafgifterne sænket, hvilket resulterede i, at den maksimale registreringsafgift i 2015 blev sænket fra 180% til 150%. I 2017 blev regeringen VLAK og Dansk Folkeparti enige om yderligere lettelse på registreringsafgiften, hvor blandt andet den lave sats i afgiften sænkes fra 105% til 85%. Derudover blev grænsen for den høje sats ændret fra 106.600,- til 185.000,-, hvilket gør de mindre biler endnu billigere. Disse tiltag tilgodeser især familier, som i forvejen har én bil, men skal bruge en ekstra, og måske lidt mindre, bil. (Finansministeriet, 2017)

I takt med at bilparken udbygges og der sker store stigninger i trafikken på de danske veje, skabes mere trængsel. I hovedstadsregionen, hvor trængslen generelt er værst, vurderes den samfundsmæssige værdi af trængselstimerne til

at have en værdi på mere end otte milliarder kroner om året. Den øgede trængsel resulterer i forsinkelser for både pendler- og erhvervstrafikken. (COWI and Finansministeriet, 2012) Der investeres massivt i både nye infrastruktur anlæg og udbygninger af eksisterende anlæg. Når der bygges ny infrastruktur i sammenspil med eksisterende, vil der opstå tidspunkter, hvor konsekvenserne af ombygningen kan være ugenomsigtige. Blandt andet ved etableringen af et projekt i milliardklassen som eksempelvis Århus letbane, kan det være svært at gennemskue konsekvenserne for den generelle trafik i Århus, når det kræver lukninger og omstruktureringer af flere veje. I takt med at trafiksystemet bliver presset til bristepunktet, er det politikernes opgave at finde bedre løsninger, som kan sikre et højt serviceniveau på de danske veje.

### 1.1 TRAFIKMODELLERING

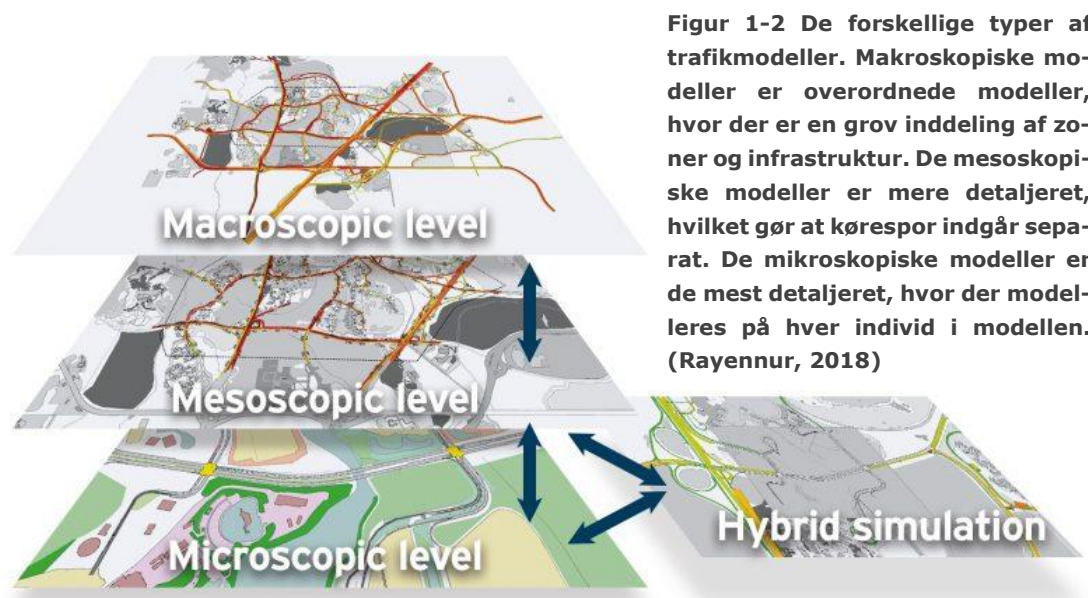
Til at gennemskue konsekvenser og belyse mulige udfald for de forskellige løsningsforslag, som bliver udarbejdet, har trafikmodellerne vundet indpas i den

gængse trafikplanlægning. I dag bruges trafikmodellerne blandt andet til at opbygge forskellige scenarier og løsningsforslag, hvilket har til formål at belyse konsekvenserne ved eksempelvis ombygninger af infrastruktur anlæg. Trafikmodeller opdeles generelt i tre forskellige grupper:

- Makroskopisk model
- Mesoskopisk model
- Mikroskopisk model

De makroskopiske modeller kan, med udgangspunkt i den nuværende trafiksituation, benyttes til at belyse den fremtidige trafiksituation. Denne type af trafikmodel arbejder overordnet med regional og kommunal planlægning, hvilket også er afspejlet i det lave detaljeringsniveau.

På samme måde som de makroskopiske modeller, danner de mesoskopiske modeller et overblik over trafikstrømme på et overordnet niveau, dog med en større detaljeringsgrad end de makroskopiske modeller. De benyttes derfor oftest til at vurdere konsekvenserne ved ændringer



**Figur 1-2 De forskellige typer af trafikmodeller. Makroskopiske modeller er overordnede modeller, hvor der er en grov inddeling af zoner og infrastruktur. De mesoskopiske modeller er mere detaljeret, hvilket gør at kørespor indgår separat. De mikroskopiske modeller er de mest detaljeret, hvor der modelleres på hver individ i modellen. (Rayennur, 2018)**

i infrastrukturen såsom et nyt boligområde eller andre større ændringer. Fælles for både de makroskopiske og mesoskopiske modeller er, at de arbejder med gennemsnitsværdier, hvor forsinkelser og kølængder beregnes gennemsnitligt for hver strøm.

Ved de mikroskopiske modeller øges detaljeringsgraden markant, hvor hver enkelt trafikant modelleres og indgår som et selvstændigt objekt i modellen. Mikroskopiske modeller benyttes oftest til at

modellere små og mere afgrænsede trafiktekniske tiltag i blandt andet byområder, inden de implementeres i virkeligheden. Da den mikroskopiske model opbygger modellerne på individniveau, bestræbes der mod, ved hjælp af de rigtige indstillinger og data, at skabe en realistisk gengivelse af virkeligheden indenfor modellens rammer. (Lahrmann and Leleur, 1994; PTV Group, 2016)

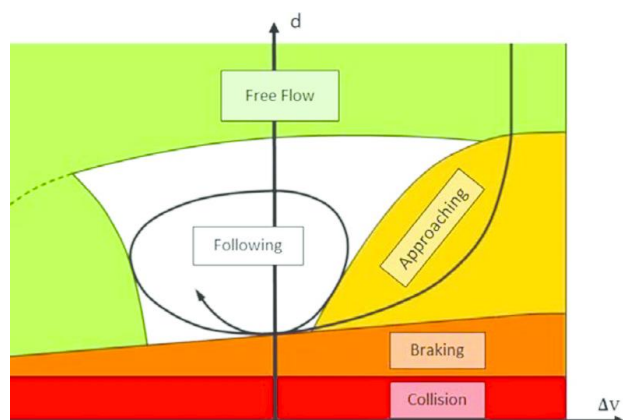
Som tillæg til de mesoskopiske modeller og mikroskopiske modeller, findes også

hybrid simuleringer, hvor dele af simuleringen foregår på henholdsvis mikroskopisk- og mesoskopisk niveau. Det kan være ved mindre systemer, hvor der er behov for en skærpet detaljeringsgrad i enkelte kryds.

Fælles for alle modellerne er, at resultater er afspejlet i det data, som modellerne fodres med. Hvis modellerne opbygges med forældet eller utilstrækkelig data, vil resultatet være tilsvarende. Hvis der ikke er tilstrækkeligt data benyttes grundparametre, som ikke altid afspejler virkeligheden på den pågældende lokalitet. Tiden, der bliver brugt til at opbygge, kalibrere og validere modellerne har også stor betydning, da kvaliteten ofte er afspejlet i den mængde tid eller penge, som er lagt i det enkelte projekt. De fleste programmer, som kan opbygge trafikmodeller, er designet og programmeret i udlandet, og det er sjældent transparent, hvad der foregår i de bagvedliggende algoritmer. Det kan være problematisk, da de danske trafikanters adfærdsmønstre ikke nødvendigvis lig de adfærdsmønstre, som optræder i andre europæiske lande.

## 1.2 MIKROSIMULERING

Programmet VISSIM, som er det mest anvendte simuleringsprogram til mikroskopiske modeller i Danmark, har mulighed for at ændre på forskellige parametre. Det gælder både adfærdsparametre for den enkelte bilist, såsom afstand til forankørende og acceleration fra stilstand, men også mere stedspecifikke parametre, såsom hastigheder i sving og det tidsrum, som vigende trafikanter skal bruge, før de vil køre ud i krydset. VISSIM modellerer adfærd ved hjælp af Wiedemanns psykofysiske ad-



**Figur 1-3 Wiedemanns car-following model.**  
(Motamedidehkordi, Margreiter and Benz, 2016)

færdsmodel, som ofte betegnes forfølgelsesmodellen (car-following). Der er grundlæggende fire forskellige tilstande, som vist på figur 1-3. (PTV Group, 2016)

**Fri kørsel:** I denne tilstand kører trafikanten frit uden påvirkning af andre trafikanter.

**Tilnærmelsestilstand:** Trafikanten nærmer sig den langsommere forankørende og decelerer derfor indtil sikkerhedsafstanden er nået.

**Forfølgelsestilstand:** Tilstanden indfinder sig, når sikkerhedsafstanden mellem trafikanten og den forankørende er konstant. I denne tilstand vil hastighedsforskellen oscillere omkring nul mellem de to bilister.

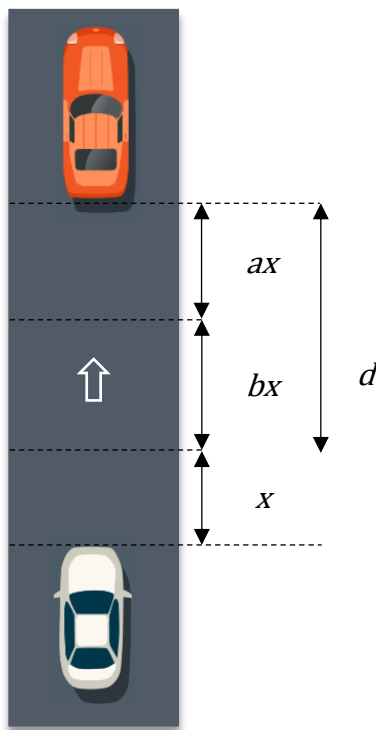
**Decelerationstilstand:** Trafikanten decelerer kraftigt, da afstanden til forankørende er lav. Denne tilstand sker oftest ved vognbaneskit eller kødannelse.

Wiedemann har udviklet to modeller, som er baseret på hans forfølgelsesprincipper. Den ene model, Wiedemann 74,

bruges primært i forbindelse med bytrafik og krydsninger eller flettestrækninger. Den anden model, Wiedemann 99, bruges primært til motorvejskørsel, hvor der ikke er sammenfletninger. Da det er to uafhængige modeller er der forskellige parametre til hver model. (PTV Group, 2016) Forfølgelsesmodellerne tilføjes i VISSIM til enkelte vejdele, hvilket gør det muligt at have en model med begge typer forfølgelsesmodeller alt efter, hvor trafikanterne bevæger sig i den førnævnte model. Det gælder især modeller, hvor motorveje indgår, da de oftest benytter Wiedemann 99 til motorvejskørsel og Wiedemann 74 til flettestrækninger og i forbindelse med krydsninger.

Forfølgelsesmodellen Wiedemann 74, som benyttes til by og flettende trafik, består af tre primære parametre: *average standstill distance [m]*, *additive part of safety distance [-]* og *multiplicative part of safety distance [-]*. De tre parametre bruges til at beregne sikkerhedsafstanden mellem to biler,  $d$ , som fås af følgende formel:





**Figur 1-4** Forfølgelsesteori fra Wiedemann 74. *ax* beskriver den gennemsnitlige afstand for holdende biler. *bx* beskriver sikkerhedsafstanden til forkørende ved kørsel, som er afhængig af blandt andet bilens hastighed. *x* er et tillæg til sikkerhedsafstanden, som bliver lagt på i visse tilfælde fx ved Conflict Areas.

$$d = ax + bx$$

Hvor:

*ax*: average standstill distance [m]

$$bx = (bx_{add} + bx_{mult} \times z) \times \sqrt{v}$$

Hvor:

*bx*: safety distance [m]

*bx<sub>add</sub>*: additive part of safety distance [–]

*bx<sub>mult</sub>*: multiplic. part of safety distance [–]

*v*: hastighed [m/s]

*z*: variabel med range [0,1], som er normal – fordelt omkring 0,5 med standardafvigelse på 0,15.

Flere studier viser, at netop disse parametre har en markant betydning for VISSIM modellerne. I (Siddharth and Ramadurai, 2013) finder de frem til, ved hjælp af følsomhedsanalyse, at alle tre parametre, *ax*, *bx<sub>add</sub>* og *bx<sub>mult</sub>*, er sensitive. I (Lu *et al.*, 2016) findes *bx* også signifikant sammen med hastigheden. (Ištoka Otković, Tollazzi and Šraml, 2013a) undersøger også de tre adfærdsparametre og finder *bx<sub>add</sub>* til at være mest signifikant, men understreger samtidig, at de alle har indflydelse på simuleringerne, da de er afhængige af

hinanden. (Li *et al.*, 2013) viser også, at adfærdsparametrene har en signifikant betydning for modellerne, hvor *bx<sub>add</sub>* igen har den største betydning. Samtidig indgår også parametre i forhold til hvor langt den enkelte trafikant kan se frem og tilbage i modellen samt hvor mange objekter den enkelte trafikant kan håndtere på samme tid. Disse parametre er nævnt til at gøre en signifikant forskel.

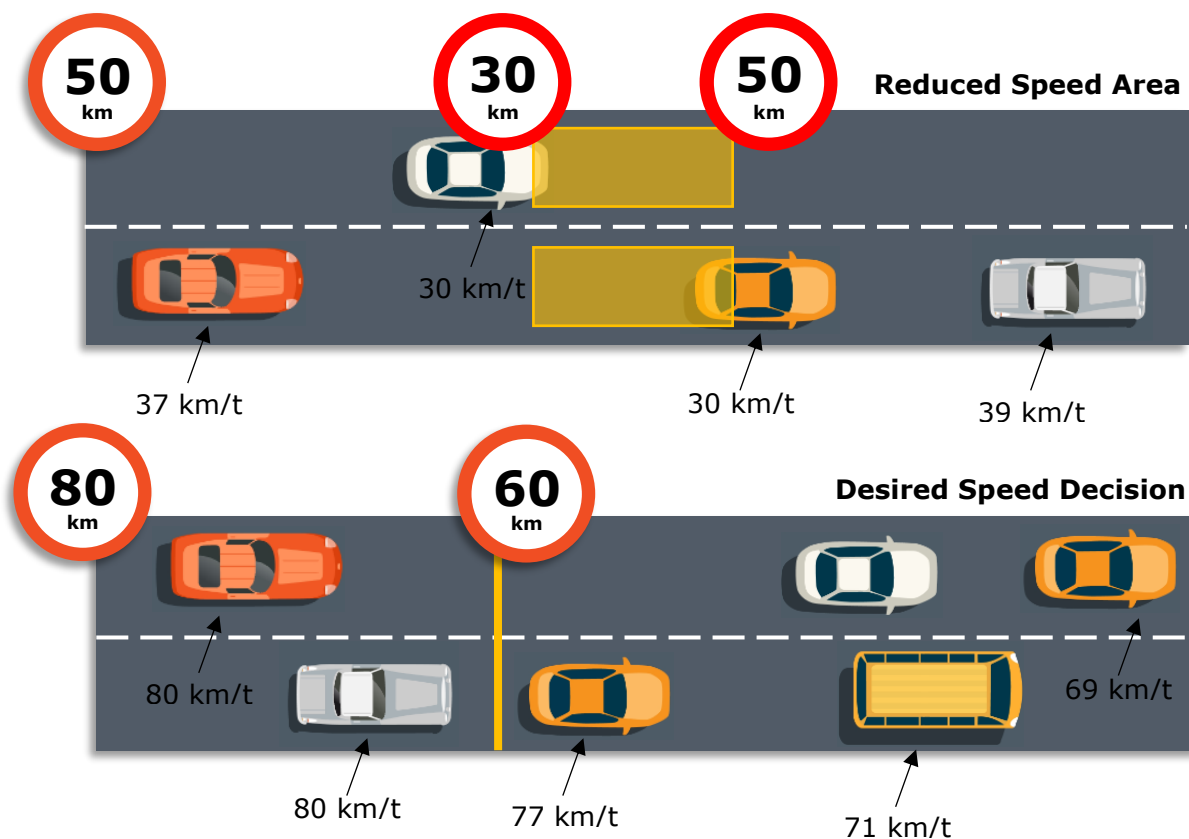
I forhold til modelopbygning i VISSIM tages der udgangspunkt i plantegninger eller ortofotos, hvor information om vej-længder, vejbaner og andre restriktioner fremgår. Dertil er information om hastigheder og svingbevægelser samt trafikvolumen også en nødvendighed for at opbygge modellen. (Hellinga, 1998)

I VISSIM er der grundlæggende to metoder til modellere hastighedsændringer. Den ene metoder er *Desired Speed Decision*, som hovedsageligt benyttes til at modellere den skilte hastighed. Der kan differentieres mellem køretøjstyper og samtidig kan der lægges forskellige

fordelinger ind i forhold til hastighed og køretøjstype. I VISSIM fungerer *Desired Speed Decision* således, at der placeres et snit, hvor den valgt hastighed er gældende fra. Trafikanten sænker først sin hastighed i det øjeblik, hvor snittet passerer, som det fremgår af Figur 1-5.

Den anden metode til hastighedsbestemmelse er *Reduced Speed Area*, hvor der i VISSIM indlægges et område, hvor hastigheden skal være det angivne. I modsætning til *Desired Speed Decision* justerer trafikanter farten, så de har opnået den ønskede hastighed, så snart de kører ind i området. Dette er den mest anvendte metode, til at modellere hastighedsændringer i forbindelse med svingbevægelser og rundkørsler. Et eksempel på dette, er vist på figur 1-5.

Udover adfærdsparametrene, som ændrer adfærden for alle trafikanter, findes der også stedspecifikke adfærdsparametre i VISSIM. De adskiller sig ved kun at ændre adfærden lokalt. Det er blandt andet gældende for VISSIMs metode til at modellere vigepligt, hvor der kan ændres på, hvor aggressivt trafikanterne



Figur 1-5 Den øverste illustration er en principskitse af Reduced Speed Areas fra VISSIM. Den nederste illustration er en principskitse af Desired Speed Decisions ligeledes fra VISSIM.

kører ud forskellige steder i modellen. Selve metoden til at modellere vigepligt er baseret på tidsgab-teori. Det betyder, at der skal være en mindste tidsafstand mellem de kørende i primærstrømmen før den vigeende trafikan vil køre ud. I

praksis vil hver trafikant foretage valg om hvorvidt de ønsker at acceptere gap'et (og dermed køre ud) eller forkaste gap'et (og dermed vente på et større gap). Det gap som trafikanter forlanger mindst skal være tilstede før de vil køre ud betegnes som det kritiske interval.

VISSIM har to metoder til at modellere vigepligt: Conflict Areas og Priority Rules. Priority Rules er en simpel tilgang til vigepligtmodellering, hvor der kun indgår et mindste gap før trafikanten vil køre ud og en længde hvor kørebanen skal være fri for kø. Priority Rules var den første metode, der blev udviklet til at modellere vigepligt, men i 2010 blev Conflict Areas tilføjet til VISSIM. Den er mere enkel at opsætte og giver en mere rationel opførsel hos de enkelte trafikanter. Selve opsætningen af Conflict Areas er lettere end ved Priority Rules, da VISSIM selv viser de steder, hvor der kan være konflikter mellem vejgrene. Herefter vælges hvilket strøm, der har forkørselsret og hvilken der har vigepligt.

Conflict Areas har flere muligheder for kalibrering, hvorfor det er den anbefalede metode af udviklerne. Metoden er fra 2010, hvilket gør, at der ikke er nogen forslag til værdier i vejreglen for mikrosimulering. I det følgende vil de vigtigste parametre for simulering med Conflict Areas blive gennemgået (PTV Group, 2016):

- **Visibility [m]** (synlighed) er maksimumafstanden for, hvornår trafikanterne opfatter hinanden.
- **Front Gap [s]** er det minimum gap i sekunder mellem bagenden af bilen på primærvejen og forenden af bilen på sekundærvejen, hvor der er vigepligt. Det sørger for, at den vigepligtige bilist ikke kører for tidligt ud i krydset.
- **Safety Distance Factor [-]** er en værdi, som multipliceres med sikkerhedsafstanden, som fastsættes ved den generelle kørselsadfærd.
- **Additional Stop Distance [m]** er den afstand, som bilisterne på sekundærvejen holder fra konfliktområdet. På er den imaginære stoplinje



Figur 1-6 Illustration af Conflict Areas i VISSIM, hvor det grønne område indikerer, at den ligeudkørende har forkørselsret. Trafikant til venstre har dermed vigepligt.

markeret, hvortil afstanden kan forlænges.

- **Avoid Blocking [%]** er en procentvis andel af bilerne på primærvejen, som vil holde tilbage, hvis der opstår kø på sekundærvejen. Denne sættes normalt til 0%.

For at sikre en virkelighedstro simulering kalibreres og valideres modellen, indtil den findes tilfredsstillende i forhold til, hvad der er observeret. I kalibreringsprocessen indsættes derfor det data, som er til rådighed. Hvis der mangler data fra enkelte parametre kan vejreglens anbefalinger anvendes. Det kan være alt fra hastigheder og accelerationer til afstand til forankørende. I vejreglen *"Anvendelse af mikrosimuleringsmodeller"* står følgende om kalibreringsprocessen:

*"Kalibreringsprocessen er derfor en iterativ proces, hvor modelbyggeren gradvist justerer de enkelte dele af modellen ind, så der opnås bedst mulig overensstemmelse med observationerne for den virkelige trafik." (Albrink et al., 2010)*

Hvor der i kalibreringsprocessen fokuseres på enkelte parametre i modellen, går valideringsprocessen ud på at vurdere modellen i forhold til virkeligheden. Der er således tale om hele modellens korrekthed, og ikke blot enkeltstående parametre. Her arbejdes der med kalibreringsmål (**M**eaure **O**f **P**erformance). Der kan i valideringsprocessen blandt andet tages udgangspunkt i følgende kalibreringsmål (Hellinga, 1998):

- Kølængder
- Rejsetid
- Trafikintensitet ved modellens afgrænsninger.

Hvis modellen ikke godkendes i valideringsprocessen, skal der derfor ændres på parametrene i kalibreringsprocessen. Det skal dog nævnes, at der kan opstå situationer, hvor modellens kølængder og rejsetider passer, men modellen kan stadig være forkert kalibreret. Kalibreringsprocessen og valideringsprocessen stræber mod, at skræddersy modellen så den passer til den virkelighed, der opleves i dag.

Til trods for at modellerne har til formål at belyse fremtidige tiltag, bliver modellerne kalibreret efter virkeligheden, som den er i modelåret. Herefter ombygges eller fremskrives modellerne, så det passer i forhold til, hvilke planer der er for fremtiden. Det er alt fra fremskrivninger 20 år frem til ændringer af den fysiske udformning på vejanlægget eller en kombination af begge. Hvis trafik anlægget bliver mere belastet, vil trafikanterne højst sandsynligt opføre sig anderledes end først antaget i forbindelse med kalibreringen. Det kan være adfærd såsom aggressivitet, og dermed villighed til at køre ud, eller den enkelte trafikants acceleration. Ændringer i adfærd afhængigt af hvor belastet systemet er, tager modellerne ikke højde for, medmindre der bliver ændret i den tidligere kalibrerede model. Den primære fremgangsmåde i dag, som også er den fremgangsmåde, der er beskrevet i vejreglen, er en iterativ proces, hvor modelbyggeren selv ændrer på parametrene indtil modellen er tilfredsstillende. (Albrink et al., 2010)

### 1.3 MASKINLÆRING OG OPTIMERING

I takt med at vores computersystemer bliver stærkere og bedre til at håndtere data, og idet data i dag er mere tilgængeligt end nogensinde, har computerbranchen de seneste år blandet andet haft fokus på emnet "maskinlæring" og optimeringsprincipper. Maskinlæring handler om at give computeren evnen til at lære eller finde sammenhæng i data uden, at de er specifikt programmeret til det.

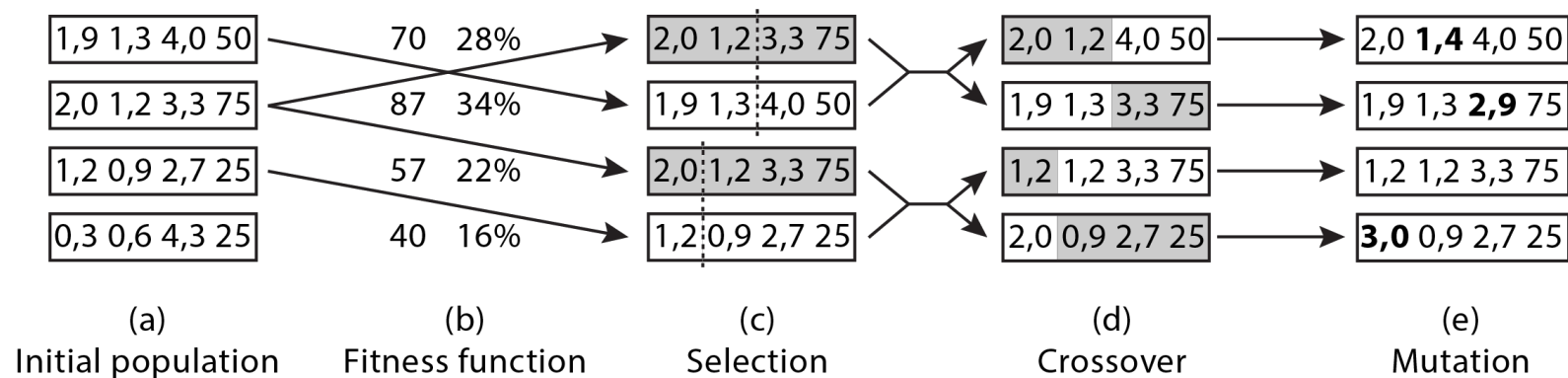
Der arbejdes der primært med to typer af maskinlæring: hjulpet og uhjulpet maskinlæring. Forskellen mellem hjulpet og uhjulpet maskinlæring er, at ved hjulpet maskinlæring er der givet et prædefinerede svar eller labels. Hvis der derfor eksempelvis er givet et sæt med prædefinerede hastighedsfordelinger for personbiler, lastbiler og cykler, så vil maskinlæringsalgoritmen kunne håndtere rå datasæt med henblik på at kategorisere datasættene. Den uhjulpet maskinlæring vil blot være i stand til at analysere og kategorisere rådata. Fordelen

ved maskinlæring er, at den kan processere enorme mængder af data, og træffe valg på baggrund af prædefinerede instruktioner. (Bishop, 2006; Norvig and Russel, 2009)

Maskinlæring og optimeringsprocesser kan derfor være med til at løse den iterative kalibreringsproces, som er den gængse metode i dag. Ved at danne et kalibreringsmål vil algoritmen være i stand til at ændre og evaluere på modellerne langt mere automatiseret end modelopbyggeren er i stand til.

Der findes flere forskellige metoder til at komme dette problem til livs, men den simpleste metode er, at skabe en algoritme som tester alle mulige scenarier i forbindelse med kalibreringsprocessen. Allerede her kan processen optimeres ved at definere intervaller, som algoritmen må variere løsningerne indenfor. Denne metode er krævende både tidsmæssigt, men også computermæssigt, da algoritmen ikke forbedrer sine gæt gang for gang, men blot kører alle løsninger igennem, hvorefter den til sidst evaluere på alle løsningerne.

En anden, og mere kompliceret, metode er såkaldt "Genetiske algoritme", hvor algoritmen forsøger at søge hen imod et allerede defineret distancemål. Denne metode er den mest anvendte metode i forhold til automatisk kalibrering af trafikmodeller. (Kim, 2005; Park and Qi, 2005; Park and Jongsun Won, 2006; Ištoka Otković, Tollazzi and Šraml, 2013b; Siddharth and Ramadurai, 2013; Tettamanti *et al.*, 2015; Yu and Fan, 2017) Genetiske algoritmer er baseret på Charles Darwins teori om evolution, hvor de stærke gener overlever og bliver videreført i takt med evolutionen. I dette tilfælde vil de bedste værdier ud fra en given population bliver benyttet til at skabe den næste population. Denne proces fortsætter indtil populationen konvergerer med den næste population og der er dermed ikke en signifikant forskel mellem de to populationer. Der er grundlæggende fem faser i en genetisk algoritme (Norvig and Russel, 2009):



Figur 1-7 Proces for Genetiske Algoritmer.

**Initial population:** Processen begynder med en udvalgt populationsstørrelse, som består af et antal mulige løsninger. I dette tilfælde et antal forskellige kombinationer af parameteropsætninger, hvor den rigtige parameteropsætning kan være indeholdt i. Et eksempel kunne være fire parameteropsætninger med fire forskellige parametre, som vist på figur 1-7.

**Fitness function:** For at kunne udvælge de løsninger, som skal arbejdes videre med, opsættes der en fitness funktion, som beskriver, hvor god løsningen er i forhold til distancemålet. I

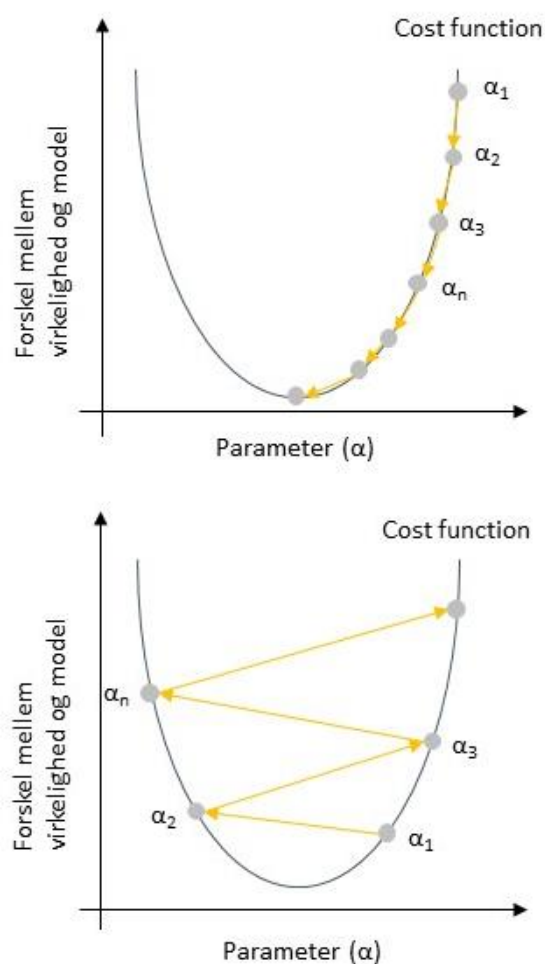
praksis vil det være en funktion, som beskriver forskellen mellem løsningen og det mål, der arbejdes hen imod. På figur 1-7 viser tallene en score, som hver parameteropsætning får og derefter en sandsynlighed for, at de bliver bragt videre.

**Selection:** I denne fase udvælges de løsninger, som er bedst i forhold til fitness funktionen. De bedste løsninger bliver derefter koblet tilfældigt sammen i par baseret på deres fitnessscore. Der udvælges samtidig tilfældige crossover points, som vist på figur 1-7.

**Crossover:** Her foretages den såkaldte crossover, hvor løsningerne bliver splittet og får byttet dele af deres værdier. Denne fase kan give store ændringer og eventuelle forværringer i forhold til tidligere population, men i takt med at algoritmen kører igennem flere gange, vil der ændringerne bliver mindre og nemmere at processere.

**Mutation:** Den sidste fase har muligheden for at mutere en tilfældigt tal i en af de nye skabte løsninger. Dette gøres for at skabe en tilfældighed i population og sikre, at der ikke blive skabt en for tidlig konvergens. De nye muterede løsninger erstatter herefter de dårlige løsninger fra





**Figur 1-8** Den øverste illustration viser, hvordan hver iteration langsomt nærmer sig et lokalt minimum. For små steps gør algoritmen langsomt at køre. Den nederste illustration viser, hvordan for store steps kan ramme forkert og aldrig finde minimum.

tidligere og bliver koblet sammen med de tidligere udvalgte løsninger, hvorefter processen kører igen.

Denne proces køres gentagende gange indtil stop kriteriet mødes. Stop kriteriet er typisk baseret på distancemålet, således at algoritmen optimerer indtil det er nået, men i visse tilfælde kan stop kriteriet også være afhængig af antal simuleringer.

Udover genetiske algoritmer kan metoden *Gradient Descent* benyttes. I modsætning til genetiske algoritmer, benytter *Gradient Descent* de tidligere resultater aktivt til at bevæge sig i den rigtige retning. *Gradient Descent* er meget matematisk tungt, men giver mulighed for en mere systematisk tilgang til at finde den optimale parameterindstilling. Metoden er yderst afhængig af, hvor store iterationssteps, der bliver inkorporeret i algoritmen. Hvis der tages for små iterationssteps vil algoritmen blive utrolig langsom, da den regner for mange scenarier ud. Hvis der derimod bruges for store iterationssteps, som vist på figur

1-8, vil algoritmen aldrig finde minimum, da den vil blive ved med at ramme langt forbi.

*Gradient Descent* er mere kompleks end genetiske algoritmer og kræver generelt ikke så meget tid eller computerkraft for at ramme et minimum. Der er dog en risiko for, at *Gradient Descent* rammer et lokalt minimum frem for et globalt minimum. Det kommer ikke til udtryk på figur 1-8, hvor der kun er vist et minimum, men i situationer hvor der er flere lokale minimum, vil det være muligt at algoritmen finder ned i et lokalt minimum, som ikke er det globale minimum, og dermed den bedste løsning. Dette sker dog oftest ved komplicerede modeller, hvor der indgår mange parametre i analysen.

Der foreligger allerede studier og artikler, hvor der er benyttet maskinlæring til at foretage følsomhedsanalyser og kalibrering af VISSIM modeller. En gennemgang af de brugte artikler kan ses i bilag A og i det efterfølgende afsnit. Disse studier er forsøget primært at kalibrere Wiedemanns forfølgelsesparametre,

men i et tidligere projekt har jeg fundet ud af, at stedspecifikke parametre, såsom vigepligtsregulering, er yderst følsomme, og dermed betyder meget for trafikafviklingen i modellen. Det gør det yderst relevant at inddrage netop disse parametre i en automatiseret proces, hvor målet er, at skabe en virkelighedstro model. (Yde-Nielsen, 2018)

Ved de tidligere studier er det forskelligt hvilket distancemål, kalibreringsmål, som er benyttet til kalibreringen, og især indsamlingen af denne type data, er meget forskelligt. COWI har i samarbejde med en ekstern partner udviklet analyseværktøjet DataFromSky, som bruger maskinlæring til at detektere og adskiller biler fra omgivelserne, og dermed tracke trafikanter igennem den optagede dronevideo. Det giver data omkring den enkelte trafikant, som tidligere har været stort set umulige at fremskaffe, og netop disse informationer er yderst brugbare i forhold til kalibrerings- og valideringsprocesserne indenfor mikrosimulering. Det gør specielt metoden til at indsamle virkelige data, som modellerne kan cali-

breres op imod, nemmere og mere præcist end de tidligere mere omstændige metoder, som er anvendt i det læste litteratur. (RCE SYSTEMS, 2017)

#### 1.4 OPSAMLING AF LITTERATUR-SØGNING

Grundlæggende viser litteratursøgningen, at de primære kalibreringsparametre, som anvendes i VISSIM er adfærdsparemetre (Driving Behavior). Alt efter om der modelleres byområder (Wiedemann 74) eller motorvejskørsel (Wiedemann 99) benyttes de respektive adfærdsmødelles, men grundlæggende er det de primære kalibreringsparametre. Dertil er der for visse artikler også anvendt hastighedsfordelinger og acceleration/deceleration. Tabel 1-1 indeholder et mindre review af relevante artikler fra litteratursøgningen, hvor optimeringsmetode, kalibreringsparametre, measures of performance og netværkstype er angivet.

Litteratursøgningen har primært været koncentreret om automatisk kalibrering

af VISSIM mødelles, og det fremgår tydeligt af tabel 1-1, at den mest anvendte metode til netop dette er med genetisk algoritmer (GA). Dertil er der også et par følsomhedsanalyser, som undersøger, hvor stor betydning de forskellige kalibreringsparametre har. Ligesom for resten af artiklerne har det primære fokus også været omkring adfærdsparemetrene.

Størstedelen af de læste artikler benytter rejsetid, som kalibreringsmål (measure of performance), da det er relativt nemt at indsamle virkelige data til. Samtidig benyttes også hastigheden i udvalgte punkter som kalibreringsmål, da også er relativt nemt, at indsamle data for. Det er dog primært ved motorveje og større netværk, at hastigheden er brugt.

| Navn                                                                                                                                                                  | Optimeringsmetode            | Measure of Performance                 | Kalibreringsparametre                                                       | Netværkstype                                |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------|----------------------------------------|-----------------------------------------------------------------------------|---------------------------------------------|
| Development and Evaluation of a Procedure for the Calibration of Simulations Models ((Brian) Park and [Maggie] Qi, 2005)                                              | GA                           | Rejsetid                               | Priority Rules (vigepligt)<br>Driving Behaviors W74<br>Hastighedsfordeling  | Signalreguleret kryds                       |
| Calibration of Microsimulation Models Using Nonparametric Statistical Techniques (KIM, 2005)                                                                          | GA                           | Rejsetid                               | Driving Behaviors W74                                                       | Strækning med flere signalregulerede kryds. |
| Microscopic Simulation Model Calibration and Validation: Case Study of VISSIM Simulation Model for a Coordinated Actuated Signal System (Park and Schneeberger, 2003) | GA                           | Rejsetid<br>Køtlængde<br>Visuelt       | Driving Behaviors W74                                                       | Større lige strækning i by.                 |
| Calibration of microscopic traffic simulation models using metaheuristic algorithms (Yu and (David) Fan, 2017)                                                        | GA<br>TS                     | Flow<br>Hastighed                      | Driving Behaviors W99                                                       | Motorvej                                    |
| Calibration of microsimulation traffic model using neural network approach (Ištoka Otković, Tollazzi and Šraml, 2013b)                                                | Machine Learning             | Rejsetid<br>Køtlængder<br>Antal stop   | Driving Behaviors w74<br>Hastighedsfordelinger                              | To rundkørsler                              |
| Iterative Calibration of VISSIM simulator Based on Genetic Algorithm (Tettamanti et al., 2015)                                                                        | GA<br>Iterativ proces        | Hastighed<br>(Floating Car Data)       | Hastighed                                                                   | Motorvejstilslutning                        |
| A video-based approach to calibrating car-following parameters in VISSIM for urban traffic. (Lu et al., 2016)                                                         | Videobaseret                 | Rejsetid                               | Driving Behaviors w74<br>Hastighedsfordelinger<br>Acceleration/deceleration | Signalreguleret kryds                       |
| Calibrating VISSIM roundabout model using a critical gap and follow-up headway approach. (Li et al., 2013)                                                            | Statistisk følsomhedsanalyse | Hastighed<br>Acceleration/deceleration | Driving Behaviors w74<br>Reduced Speed Areas<br>Priority Rules              | Rundkørsel                                  |
| Analysis of the influence of car-following input parameters on the modelled travelling time. (Ištoka Otković, Tollazzi and Šraml, 2013a)                              | Følsomhedsanalyse            | Rejsetid                               | Driving Behaviors w74<br>(AX, BX <sub>add</sub> og BX <sub>mult</sub> )     | Rundkørsel                                  |
| Microscopic Simulation Model (VISSIM) Calibration for Freeways with Managed Lanes (Yang and Mattingly, 2011)                                                          | GA                           | Rejsetid<br>Hastighed                  | Driving Behaviors w99<br>Acceleration/deceleration                          | Motorvej                                    |

Tabel 1-1 Review af relevante observationer fra litteratursøgningen.

## 1.5 PROJEKTLOKALITET

I projektet tages der udgangspunkt i en rundkørsel ved motorvejsafkørslen ved Taulov mellem Fredericia og Kolding, som vist på figur 1-9. Rundkørslen er allerede tracket af COWIs dronesoftware, hvilket gør det muligt, at foretage analyser i forbindelse med kalibreringen af VISSIM modellen. Der er valgt en rundkørsel, da flowet er afhængig af trafikanternes interageren hinanden imellem i forhold til vigepligt. Rundkørslen består af en frakørsel og tilkørsel til motorvejen samt tre andre vejgrene. Det vestlige ben, Adelvej, er den direkte vej ind til Taulov by, hvor der bor omtrent 3400 indbyggere (Danmarks Statistik, 2018). Det nordlige ben, Skærbækvej N, fører op til et større industriområde og er samtidig opkoblingen til motorvejen i nordlig retning. Det sydlige ben, Skærbækvej S, fører ned til Arla Foods Mejeri og videre til Kolding Landevej. Industriområdet nord for rundkørslen og Arlas Mejeri generer en del trafik til rundkørslen samtidig med at pendlertrafikken fra Taulov belaster systemet. Rundkørslen



Figur 1-9 Projektlokalitet. ÆNDRE BILLEDE!

har en  $\emptyset$ -diameter på 35 meter og cirkulationsbredder på 6,5 meter.

Den primære trafik, i morgenspidstimen, kommer fra Adelvej og motorvejen mod Skærbækvej i nord og syd. Der er i kapitel 2 vedlagt OD matricen for krydset i morgenspidstimen, som er indsamlet ved hjælp af DataFromSky.

## 1.6 PROJEKTBESKRIVELSE

Indeværende projekt har til formål at skabe en bro mellem droneoptagelsernes resultater og parametrene, som optræder i VISSIM. Ved hjælp af maskinlæring og et tillægsmodul til VISSIM, COM-modul, vil det være muligt at skabe

et tillægsprogram, hvor computeren selv foretager den iterative proces og kalibrer modellerne med henblik på at ramme udvalgte mål. De mål, såkaldte distancemål eller kalibreringsmål, kan blandt andet være målbare enheder såsom forsinkelse, rejsetid og kølængder. Algoritmen, alt efter hvilken algoritme der vælges, vil selv justere adfærdsparametrene indtil forskellen på distancemålet mellem den trackede droneoptagelse og VISSIM-modellen er nær nul. Ud fra projektbaggrunden og disse betragtninger er følgende problemformulering opsat, som danner rammerne for dette projekt:

*Hvordan opsættes et tillægsprogram til VISSIM, som selv, ved hjælp af maskinlæringsprincipper, kan kalibrere modeller ud fra forudbestemte distancemål? Hvilke kalibreringsparametre bør indgå, og hvor godt modellerer VISSIM modellerne virkeligheden?*

### 1.7 PROJEKTAFGRÆNSNING

Der er i projektet kun fokuseret på kalibrering af personbiler og tunge køretøjer (lastbiler, busser og sættevogne), hvorfor bløde trafikanter ikke indgår i hverken trafiktællinger eller modellering.

I forbindelse med registrering af parametre i droneoptagelserne er det forudsat, at det data, COWI stiller til rådighed, er kalibreret korrekt. Der antages derfor, at hastigheder, accelerationer mv. er i overensstemmelse med virkeligheden. Der vil blive foretaget overfladiske screeninger for de yderste værdier, men hver enkelt værdi vil ikke blive gennemset. Denne afgrænsning er foretaget, da projektets formål er, at kalibrere modeller med trackingdata og ikke hvor pålidelig selve trackingprogrammet er.

Til kalibreringen af modellerne er der brugt MATLAB, som har en toolbox, hvor optimeringsværktøjer er i. Denne toolbox bruges i stedet for at kode egne algoritmer, da jeg ikke har den fornødne viden omkring kodning af disse.

I projektet er der fokuseret på at opbygge en automatiseret proces, der virker i praksis, hvor et færdigt produkt er vægtet højere end en kompliceret kalibrering, der teoretisk virker. De yderligere overvejelser og forbedringer af den automatiserede kalibreringsproces findes i diskussionsafsnittet i kapitel 4.





## 2 METODE

*I indeværende kapitel vil de anvendte metoder i rapporten blive gennemgået. Valget af metode er vigtigt, da analyserne og dermed resultaterne er afhængige af de valgte metoder. Med udgangspunkt i ovenstående overvejelser og valg beskrives projektets metode og processer i dette kapitel. Der vil indgå forklaring af litteratursøgning og dataindsamling herunder DataFromSky, databehandling, trafikmodellering og maskinlæring til modellering.*

### 2.1 LITTERATURSØGNING

Forud for projektet er der foretaget et litteraturstudie for at klarlægge og granske den nationale og internationale forskningsviden i forhold til VISSIM modellering og specielt kalibrering af disse modeller, for at kunne bygge videre på deres resultater og konklusioner. Litteratursøgning fokuserer på emnerne VISSIM, grundparametre i VISSIM, kalibrering og validering af VISSIM modeller og maskinlæring. De vigtigste resultater af

førnævnte litteratursøgning er sammenfattet i kapitel 1, og i bilag A er en systematisk gennemgang af artiklerne listet med et kort resumé af hver.

Til litteratursøgning er der primært brugt følgende databaser:

- Primo – Aalborg Universitetsbiblioteks søgemaskine, som søger i relevante databaser for national og international litteratur
- IEEE Xplore – Digital Library
- Science Direct
- Google Scholar

Størstedelen af litteraturstudiet er dannet ved hjælp af Primo, da den søger i diverse databaser, som Aalborg Universitet har adgang til. Udover Primo er "IEEE Xplore – Digital Library" og "Science Direct" også anvendt, som enkeltstående databaser for ikke at overse litteratur. De primære søgeord, som er anvendt til litteraturstudiet, er:

- VISSIM
- Calibration
- Validation
- Machine Learning

- Microsimulation
- Modelling
- Model
- Genetic Algorithm
- **Component Object Model**

I søgningen er der anvendt trunkeringstegn, hvilket medtager ordstammen, men godtager forskellige endelser. Derudover er der også anvendte "AND" mellem søgeordene, hvilket muliggør søgninger, hvor flere ord indgår. Der er samtidig brugt anførselstegn, da det sikrer, at ordene optræder i den rigtige rækkefølge. Et eksempel på den strategi kunne være en søgning indeholdende: VISSIM AND model\* AND "machine learning". Trunkeringstegnet ved model\* sikrer, at der i søgningerne også vil blive søgt på andre endelser/bøjninger, såsom modelling, models osv. Anførselstegnene sikrer, at der bliver søgt på den skrevne rækkefølge i søgning, machine learning, og ikke learning machine, da det ville give andre resultater.

Litteratursøgningen omfatter både forskningsrapporter, videnskabelige artikler, håndbøger og brugervejledninger.

Der er i søgningen både søgt nationalt og internationalt, dog med det forbehold, at litteraturen er skrevet på enten dansk eller engelsk.

Udover litteratur, artikler og bøger, fra litteratursøgningen er der brugt brugervejledninger fra VISSIM og DataFromSky. De bistår med baggrundsviden om de modeltekniske antagelser, modelopbygning samt dataudtræk. Selve metoden til modelopbygningen og den generelle brug af programmerne vil ikke blive beskrevet i indeværende rapport, hvorfor der henvises til disse vejledninger, hvis der ønskes yderligere information.

## 2.2 DATAINDSAMLING

I projektet er der taget udgangspunkt i en rundkørsel ved motorvejsafkørslen ved Taulov mellem Fredericia og Kolding. En nærmere beskrivelse af projektlokaliteten findes i kapitel 3.

For at kunne opbygge en mikrosimuleringsmodel kræves data, som beskriver den aktuelle trafiksituation. Til dette formål er der indhentet data fra en tidligere

droneoptagelse som COWI A/S har foretaget for Vejdirektoratet i forbindelse med et studie omkring motorvejsafkørslen og de omkringliggende rundkørsler. Optagelserne er foretaget den tirsdag den 24/05-2016 om morgenen fra 7.00 til 9.00 og om eftermiddagen fra 15.00 til 17.00, da det oftest er i dette tidspunkt, hvor trafikbelastningen er størst. COWI A/S har indgået et samarbejde med RCE SYSTEMS, hvor COWI A/S optager trafikale situationer med deres egne droner, og herefter får tracket optagelserne af RCE SYSTEMS, som ejer DataFromSky. (RCE SYSTEMS, 2017)

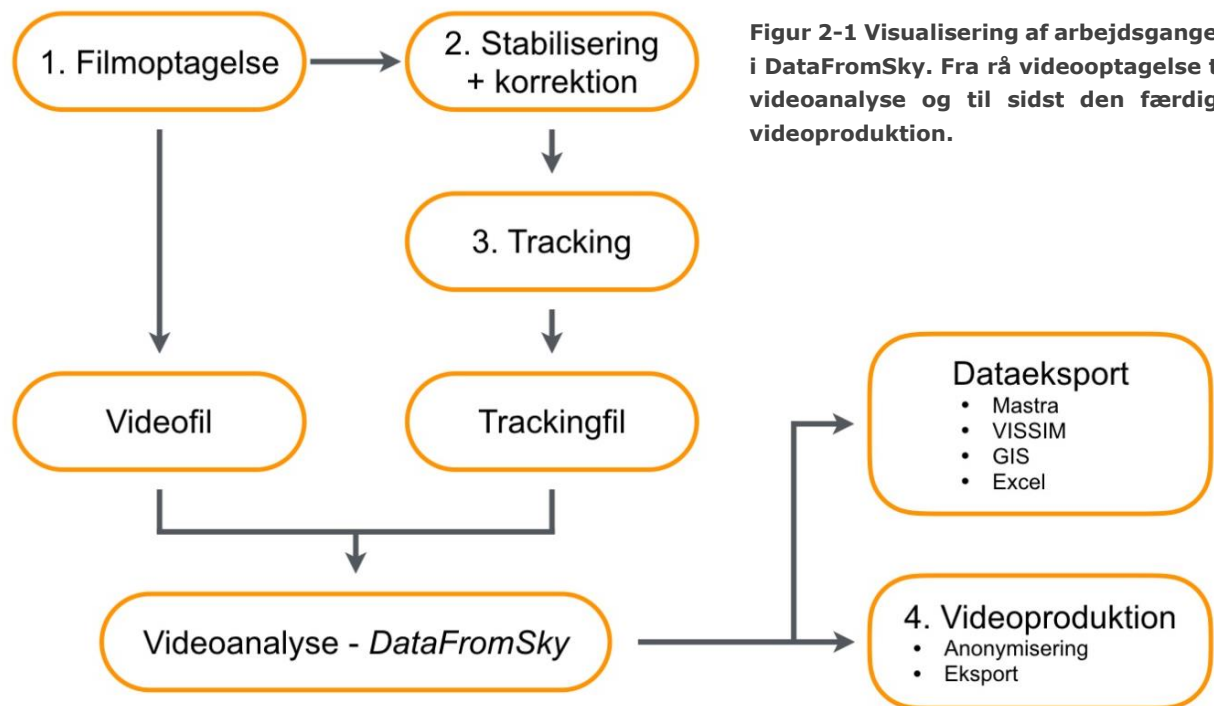
Dronerne, som bliver brugt til denne type optagelser, er udviklet af COWI. Det betyder, at de er specielt indstillet og tilpasset til formålet at optage stationære trafiksituationer i længere tid. Til de længerevarende optagelser (over 1 time) benytter COWI droner, som har en markant længere flyvetid end de traditionelle droner såsom DJI Phantom. Dertil har dronerne specielle linser, som gør det muligt at optage i høj billedkvalitet. Det er nødvendigt, da dronelovgivningen gør, at dronerne skal operere forholdsvis

langt væk fra lokaliteten, som skal optages, da det er ulovligt at flyve nær offentlige veje og byområder. (Transport-Bygnings- og Boligministeriet, 2017)

### 2.2.1 DATAFROMSKY

Med udgangspunkt i COWIs droneoptagelser tracker RCE SYSTEMS alle trafikanter i videoen. I det følgende vil grundlaget og arbejdsgangen for analyser med DataFromSky blive beskrevet. Et procesdiagram for arbejdsgangen er vist på figur 2-1.

For at selve trackingen kan foregå, bliver optagelsen geokodet ud fra dronens position samt en række referencepunkter. Referencepunkterne gør det muligt at "vrider" optagelsen ned i et plan, hvorefter der kan dannes et koordinatsystem. Selve "vridningen" er vist på figur 2-2. Når optagelsen er tvunget ned i planet, kan geokodningen foretages, hvor hver pixel får et sæt koordinater, som senere bruges til trackingen. Det gør, at hver trafikant kan stedfæstes hele optagelsen igennem, og dermed kan alle trafikanternes hastigheder og accelerationer mv. beregnes. Geokodningen muliggør også en fuld stabilisering og korrigerende af optagelsen, hvilket forhindrer billedet



Figur 2-1 Visualisering af arbejdsgangen i DataFromSky. Fra rå videooptagelse til videoanalyse og til sidst den færdige videoproduktion.

i at svaje, som typisk sker grundet eventuelt kraftig vind, som er i dronens højde.

Droneoptagelserne optages typisk med 4K-opløsning, hvilket sikrer en stor nøjagtighed, da antallet af pixels er større ved højere opløsninger. Præcisionen afhænger af, hvor langt væk fra krydset dronen er placeret. Med en 4K-opløsning

100 meter væk fra krydset har DataFromSky en koordinatpræcision på 10 cm. Alle optagelser foretages med en framerate på 25, hvilket giver data, såsom hastighed, acceleration og retning for hvert objekt 25 gange i sekundet. Med den høje framerate og opløsning kan DataFromSky antages at være relativt nøjagtig. Selve trackingen af køretøjer foregår med Deep Neural Networking

og Machine Learning, hvilket er computerbaserede algoritmer, som kan "genkende" og kategorisere billeder ud fra tidligere billeder. Selve tanken bag billedgenkendelsen er, at algoritmerne først præsenteret for en række positive og negative samples også kaldet "test set". Det kan være samples som vist på figur 2-3, hvor de er inddelt i *cars*, *vans* og *buses*. Det er vigtigt, at ens test set indeholder en stor mængde billeder, da det er hele grundlaget for den automatiske genkendelse. På figur 2-3 er der kun

vist køretøjer set forfra, men for at algoritmen kan kende køretøjer fra alle vinkler, så skal det anvendte test set også have et stort og bredt udvalg. Algoritmen regner på sandsynligheden for at det er den ene eller den anden trafikanttype (personbiler, lastbiler, busser mv.) og kategoriserer herefter hvert køretøj. I visse tilfælde hvor algoritmens beregnede sandsynlighed er for lav til at give en signifikant definition, skal der foretages en manual klassificering. (Norvig

and Russel, 2009; Babinec *et al.*, 2015; RCE SYSTEMS, 2017)



**Figur 2-2** Proces for "vridning" af videooptagelse ned i planet. Venstre billede viser stabiliseret dronevideo, højre billede viser samme billede blot vredet ned i planet.



Grunden til at denne proces kaldes for maskinlæring er, at hver gang DataFromSky har tracket en trafikant, klassificeret den og fået trackingen godkendt, indgår den i algoritmens testset for de næste trackinger. Det betyder, at hver gang der er foretaget en tracking bliver selve databasen med køretøjer større.

På nuværende tidspunkt har DataFromSky's database over 250.000 positive og negative samples og en genkendelighedsgrad på 98 % for køretøjer, hvilket kun bliver bedre i takt med, at der bliver tracket flere droneoptagelser. Da algoritmen kontrollerer hver trafikant med den store database sætter det selvsagt

krav til computerkraften, hvilket betyder, at RCE systems benytter en specialbygget computer over 2.000 kerner computerkraft.

Denne klassificering sker for hvert frame, 25 gange i sekundet, og gælder for alle objekter i optagelsen herunder køretøjer, cykler og fodgængere. Alle objekter får et unikt ID, som gør det muligt at differentiere mellem dem. For hvert objekt dannes der en såkaldt trajectory, som beskriver objektets vej igennem optagelsen. Med udgangspunkt i de dannede trajectories genereres en database, hvor hvert tracket objekt har sin egen record i databasen. Denne database rummer alle objekternes placering, hastighed, acceleration mv. til enhver tid.

DataFromSky kombinerer herefter trackingfilen og videofilen, hvilket muliggør selve databehandlingen, som vist på figur 2-1 på side 18. Denne databehandling er beskrevet i afsnit 2.3.



**Figur 2-3** Et eksempel på et testset til automatisk klassificering af køretøjer set forfra. I DataFromSky benyttes trafikanter fra flere vinkler, for at få et større testset og dermed en større nøjagtighed. Figur hentet fra (Qin, Huang and Chen, 2017).

## 2.3 DATABEHANDLING

Med udgangspunkt i analyseværktøjet DataFromSky kan der trækkes data ud vedrørende trafikanterne i den optagede rundkørsel. Data, som trækkes ud, anvendes som modelinput, hvor de benyttes i opbygningen, kalibreringen og valideringen af modellen. Der trækkes derfor følgende ud fra DataFromSky:

- OD Matrice
- Hastigheder
- Rejsetid

I følgende sektion vil metoden til disse udtræk blive gennemgået.

### 2.3.1 DATAFROMSKY

DataFromSkys databehandling består primært af såkaldte gates, som indlægges i programmet. Hvert objekt i optagelsen er gengivet ved en prik i den forreste del af køretøjet, og når den passerer de såkaldte gates, bliver informationer om trafikanterne lagret i netop dette punkt.

Disse gates tæller som udgangspunkt alle trafikanttyper, som passerer gaten og i begge retninger. I DataFromSky kan dette dog ændres til kun at omfatte forskellige retninger. På figur 2-4 er det grønne snit opsat til, at det kun er trafikanter på vej ind i rundkørslen, som bliver talt og omvendt for det røde. For hver trafikant, som har krydset gaten, gemmer DataFromSky informationer såsom Track ID, køretøjstype, tidspunkt, billede ID, hastighed, acceleration i hver retning og eventuelt tid fra forrige objekt krydsede gaten. Da hvert objekt har et

unik ID, såkaldt Track ID, og gaten lagrer krydsningstidspunkt er det muligt at sammenholde det med andre gates og derved beregne rejsetid, kølængder osv.

### 2.3.2 OD MATRICE

Med DataFromSky er det også muligt at udtrække informationer vedrørende svingbevægelser og dermed danne en komplet OD-Matrice. Der indlægges to snit i hver ben, som vist på figur 2-4. Ved at indlægge en entry og en exit gate i hvert ben, illustreret med henholdsvis grøn og rød farve, kan DataFromSky



Selected Annotation

|                      |                     |
|----------------------|---------------------|
| Annotation Type:     | Gate                |
| Gate ID:             | 39                  |
| Gate Type:           | Entry               |
| Gate Direction:      | Negative            |
| Angular Sensitivity: | 90.00 [deg]         |
| Allowed Objects:     | Select              |
| Tag:                 | Skærbækvej syd, ind |

**Figur 2-4** Venstre billede viser gates, som indsættes i DataFromSky. Højre billede viser opsætningen af gates, som kan ændres for hver gate.



selv koble Track ID op med de forskellige gates, så hver trafikants svingmønstre let kan illustreres. DataFromSky er det muligt at hive en fuld OD Matrice ud både som CSV-fil og Excel-fil.

### 2.3.3 HASTIGHEDER I CIRKULATIONSAREALET

For at finde hastighederne i cirkulationsarealet indlægges gates, der lagrer hastigheden på de cirkulerende i rundkørslen. Tidligere har jeg undersøgt cirkulationshastigheder i rundkørsler og konkluderet, at de indkørende trafikanter ikke har samme hastighed som trafikanterne, der kommer fra et af de andre ben

i rundkørslen. (Yde-Nielsen, 2018) Af den grund indsættes samtidig en gate umiddelbart inden cirkulationsarealet for at kunne differentiere mellem indkørende og cirkulerende trafikanter, som vist på figur 2-5. Da DataFromSky lagrer Track ID fra alle, som passerer gaten, er det muligt at sortere de indkørende trafikanter fra, da de vil optræde i begge gates. DataFromSkys data kan eksporteres til CSV-filer, som kan læses i Excel, hvor selve sorteringen opsættes.

Efter sorteringen kan de cirkulerende trafikanters hastighed opsættes som en akkumuleret frekvenskurve, som indsættes direkte i VISSIM.



Figur 2-5 Eksempel på sortering mellem indkørende og cirkulerende trafikanter.

### 2.3.4 REJSETIDER

For projektet vil det også være relevant undersøge rejsetiderne for forskellige ture i modellen. Dette gøres, da rejsetiden er målbar i både DataFromSky og VISSIM, hvilket kan bruges til at kalibrere efter.

Fremgangsmåden til at finde rejsetiden i DataFromSky er baseret på gates og tidsstempler. På samme måde som med OD Matricen indsættes entry og exit gates i DataFromSky, som lagrer information om de passerende trafikanter. I denne del er det Track ID og tidsstemplet, som er vigtigt, da der med udgangspunkt i de to, kan findes rejsetider for alle trafikanter. På samme måde som tidligere trækkes data ud i CSV-format og indlægges herefter i Excel til videre bearbejdning. Det er dog vigtigt, at gates i DataFromSky er placeret i samme punkt som dataindsamlingspunkterne i VISSIM, da sammenligningen forudsætter, at dataindsamlingspunkterne er ens i begge programmer.

Da DataFromSky er den begrænsende faktor rent modelstørrelsesmæssigt vil

det være nærliggende at placere rejsetidsmålerne så langt ude, som droneoptagelsen tillader. Droneoptagelsernes geokodning muliggør opmåling, hvilket bruges til at beregne afstande til stopstregen i det pågældende ben. På den måde kan der indlægges et dataindsamlingspunkt i VISSIM i samme afstand, som gaten er indlagt i DataFromSky.

Hvis der ikke har været registreret en rejsetid for et rute henover et minut kan udtrækket ikke data fra DataFromSky og VISSIM ikke sammenlignes. Derfor interpoleres der lineært mellem kendte rejsetider, for at skabe en fuld rejsetidsmatrice. I enderne, hvor der ikke kan interpoleres, benyttes det nærmeste tal. Disse omregninger er nødvendige, for at kunne sammenligne rejsetid med VISSIM, da VISSIM, grundet varierende ankomstfordeling igennem de forskellige runs, har en fyldt rejsetidsmatrice.

## 2.4 TRAFIKMODELLERING

I indeværende projekt ønskes der undersøgt mulighederne for at kalibrere VISSIM modeller med udgangspunkt i

trackede videoer. Da de trackede videoer indeholder detaljerede informationer om hver trafikant, er det nærliggende at benytte mikrosimuleringsprogrammet VISSIM som modelleringsværktøj, da det kan opnå det samme i modelleringsøjemed, som DataFromSky kan levere.

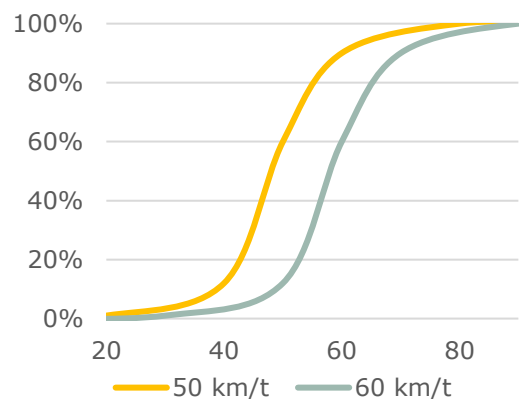
VISSIM er udviklet af PTV Group i Tyskland og er et multimodalt mikrosimuleringsprogram med en høj detaljeringsgrad. Den høje detaljeringsgrad kommer især til udtryk ved de mange parametre, som kan ændres i programmet, hvilket gør det fuldt dynamisk og giver brugeren mulighed for at skræddersy trafikantadfærd i modellen, så den er lig med den virkelig de ønsker at vise. VISSIM er et af de førende programmer indenfor mikrosimulering, og især i Danmark, hvor VISSIM stort set har monopol på al mikrosimulering. En af grundene er, at der er udarbejdet en vejregel, *Anvendelse af mikrosimuleringsmodeller*, som forklarer om principperne og opbygning af mikrosimuleringsmodeller. I vejreglen er der samtidig en række anbefalinger til standardparametre, som alle henviser til

VISSIM. Disse standardparametre medvirker til at parametrene og procedurerne bliver tilnærmelsesvis ens iblandt de forskellige rådgivningsfirmaer i Danmark.

### 2.4.1 MODELOPBYGNING

Selve modelopbygningen i VISSIM foregår på baggrund af de foreskrevne metoder i VISSIM manualen, hvor der indlægges enkelte kørespor, svingbaner og kryds. I VISSIM indlægges der ortofoto, hvilket benyttes til at danne det overordnede vejnet. Herefter indlægges hastighedsgrænser med dertilhørende opsætning af hastighedsfordelingerne. I vejreglen for mikrosimulering er der angivet hastighedsfordelinger for forskellige hastigheder baseret på reelle hastighedsmålinger, som vist på figur 2-6. (Albrink *et al.*, 2010) Disse hastighedsfordelinger indlægges i VISSIM i forbindelse med *Desired Speed Decisions*, som er gennemgået i kapitel 1.

Udover hastighedsfordelingerne i forbindelse med den tilladte hastighed ønskes også en virkelighedstro modellering af



**Figur 2-6** Eksempel på hastighedsfordelinger, som bliver anvendt i modellen.

svingmanøvre. I VISSIM er det muligt at nedsænke hastigheden lokalt i et område ved hjælp af *Reduced Speed Areas*. For en dybere forklaring af hvordan *Reduced Speed Areas* fungerer henvises der til kapitel 1. Hastighedsnedsættelserne i svingene og cirkulationsarealet er baseret på målinger fra DataFromSky. (PTV Group, 2016)

Til at modellere vigepligterne i forbindelse med rundkørslen benytte funktionen *Conflict Areas*, da det er den mest anvendte, og anbefalede, funktion til vigepligtsmodellering. VISSIM viser selv, hvor der er konflikter mellem strømme

og herefter vælges forkørselsretten. Selve opbygningen er *Conflict Areas* og de dertilhørende parametre er gennemgået i kapitel 1. Som udgangspunktet er *Conflict Areas* indstillet efter Vejreglen for mikrosimulering, men disse parametre vil blive ændret i forbindelse med kalibreringen. (Albrink *et al.*, 2010; PTV Group, 2016)

Trafikanterne rutevalg indlægges også i modellen. Som gennemgået i afsnit 2.3 benyttes DataFromSky til at danne en OD-matrice, hvor rutevalg herefter kan implementeres i DataFromSky. I VISSIM indlægges ruter fra alle vejgrene, hvor en procentvis fordeling af rutevalget indtastes. Den talte trafik indlægges i modellen på kvarters interval, hvortil den implementeres med en stokastisk ankomstfordeling, hvilket medfører at trafikanternes ankomstfordeling kan variere. Den stokastiske ankomstfordeling afspejler dag-til-dag variationen, hvilket gør at trafikintensiteten varierer fra simulering til simulering.

#### 2.4.2 MIKROSIMULERING

Efter modellen er opbygget foretages simuleringer med standardparametre og optimerede parametre, hvorefter effekten kan bestemmes. For hver parameteropsætning foretages 5 simuleringer, hvor der anvendes forskellige *random seeds*. De bevirker, at ankomstfordelingen i modellen kan variere henover de 15 minutters intervaller, hvor trafikinputtet er lagt ind. Der anvendes i projektet flere forskellige *random seeds*, for at mindske risikoen for statistisk usikkerhed.

Den enkelte simulering foretages over en to timers periode i morgenmyldretiden fra 7:15 til 8:15. Som beskrevet i vejreglen for anvendelse mikrosimuleringsmodeller benyttes en indkørselstid på 15 minutter for at sikre sig, at modellen ikke er tom, når resultatudtrækket starter. Til det benyttes resultaterne fra 7:00 til 7:15, som også er hentet fra DataFromSky. Selve OD matricen er vedlagt vist i kapitel 4.

### 2.4.3 EVALUERING

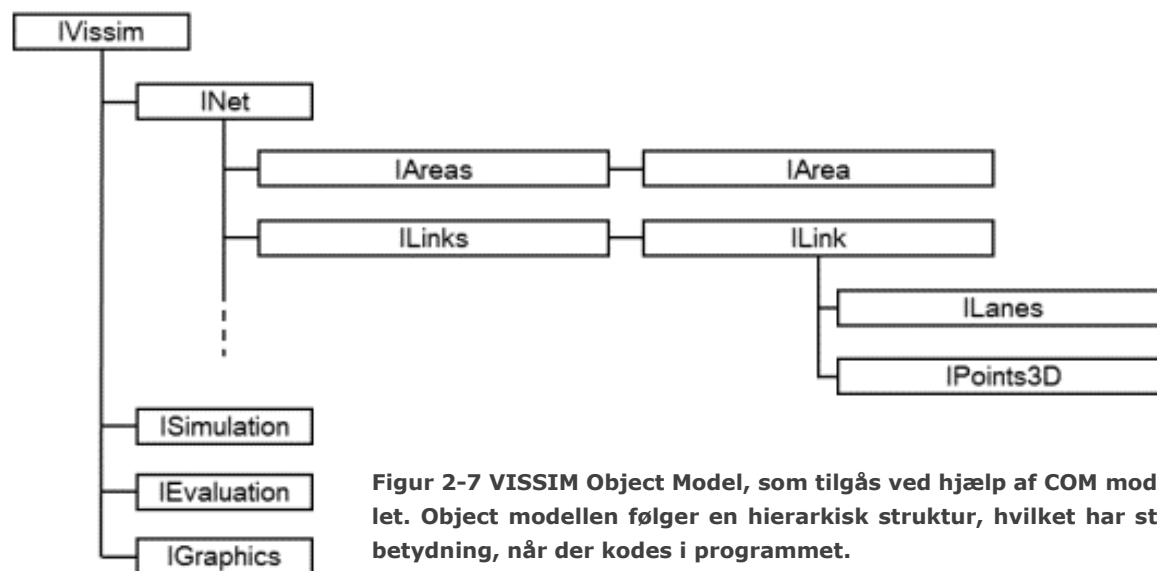
Med VISSIM er det muligt at trække resultater ud om den modellerede trafik. Det gøres ved at indlægge såkaldte *counters*, som kan samle informationer om trafikken i modellen. For rejsetid er det vigtigt, at de indlagte counters bliver lagt samme sted som i DataFromSky, da de sammenlignes i forhold til kalibreringen. I VISSIM fastsættes starttidspunkt, sluttidspunkt samt tidsintervaller for indsamling af data. Da der benyttes et indledende kvarter indstilles modellen til først at indsamle data efter 900 sekunder og frem til 4500 sekunder. Tidsintervallerne indstilles til 60 sekunder pr. interval, da det stemmer overens med intervallerne fra DataFromSky. Da der i dette projekt udelukkende benyttes rejsetid til kalibrering, er det den eneste, som er trukket ud.

### 2.4.4 COM MODUL

Den gængse tilgang til modellering med VISSIM er at opbygge, kalibrere, validere og evaluere alt direkte i VISSIM.

Det er dog også muligt at styre program-

den forrige simulering. Det vil altså være



Figur 2-7 VISSIM Object Model, som tilgås ved hjælp af COM modulet. Object modellen følger en hierarkisk struktur, hvilket har stor betydning, når der kodes i programmet.

met direkte gennem et COM (Component Object Model) modul, som trækker data direkte ud fra VISSIM. COM modulet kan tilgås ved hjælp af flere forskellige kodeprogrammer, men i dette projekt benyttes programmet MATLAB.

Grunden til at COM modulet og MATLAB benyttes, er for at kunne ændre på parametre i VISSIM og kører simuleringer ved hjælp af en automatiseret proces, så modelbyggeren ikke selv skal ændre på parametrene alt efter resultaterne fra

muligt at få MATLAB til at køre en simulering igennem for herefter at ændre parametrene i den rigtige retning, som beskrevet i afsnit 1.3.

COM modulet fungerer efter et hierarkisystem, hvor der tilgås funktioner ved at gå igennem hierarkiet. Det betyder, at for at ændre noget vedrørende *Lanes* jævnfør Figur 2-7 i programmet, skal hele stien hen til *Lanes* først skrives ned, i dette tilfælde *Vissim.Net.Links.Link.Lanes*. Der kan med COM modulet både

indhentes nuværende informationer fra programmet, men også ændre parametre og lignende direkte fra COM modulet.

For yderligere information om adgang til COM modulet og kodning af det henvises der til de respektive VISSIM manualer og håndbøger. I det elektroniske bilag er de MATLAB scripts vedlagt, som er brugt i dette projekt.

## 2.5 AUTOMATISERET KALIBRERING

Da formålet med projektet er at skabe en automatiseret proces i forbindelse med kalibrering af VISSIM modeller, bestræbes der mod at opbygge et system, som kan evaluere på resultaterne i forhold til de prædefinerede mål. Herefter skal systemet selv kunne optimere på parametrene indtil resultaterne konvergerer med de prædefinerede mål og forskellen mellem model og virkelighed er minimeret mest muligt.

### 2.5.1 COM MODUL OG MATLAB

Til at skabe en automatiseret proces benyttes MATLAB, da der kan opsættes et script, som kan køre VISSIM eksternt, hente resultater, evaluere på resultaterne og ændre parametrene. Alt dette kører i loop, indtil der er nået et tilfredsstillende resultat.

Alle optimeringsprocesserne i indeværende projekt er bygget op omkring en *fitness function*, som beskriver hvor god simuleringen er i forhold til de opsatte kriterier. I dette projekt er fitness funktionen baseret på den tidligere valgte "Measure Of Performance", som i dette projekt er rejsetid. Formålet er derfor, at få rejsetiden fra VISSIM simuleringen og rejsetiden fra DataFromSky til være så tæt på hinanden som muligt ved at ændre på parametre i VISSIM.

### 2.5.2 METODER TIL OPTIMERING

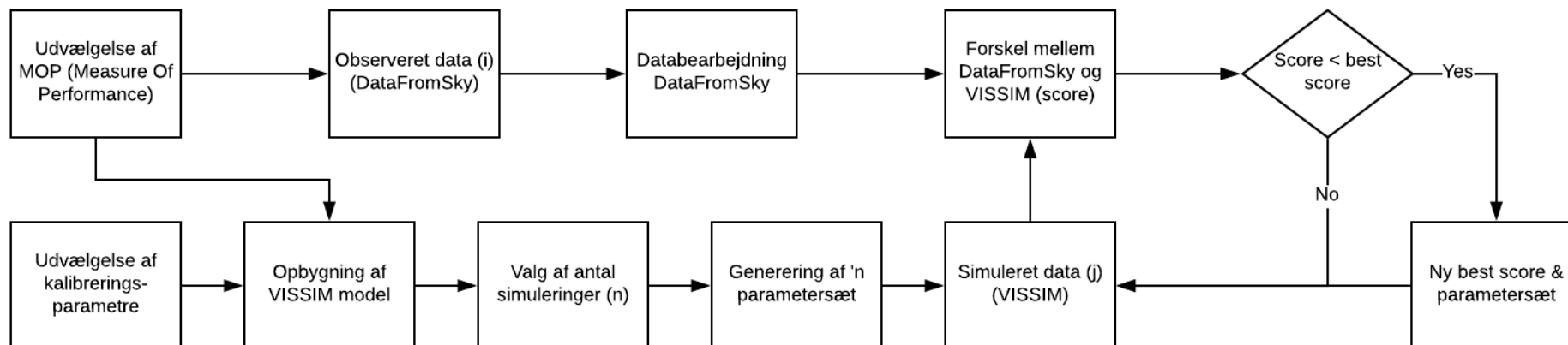
Som nævnt i tidligere afsnit er formålet, at få forskellen mellem rejsetiden i VISSIM og DataFromSky til at være lig nul. Der findes flere metoder til at løse dette optimeringsproblem. I de kommende

underafsnit vil de forskellige metoder i projektet blive gennemgået og en forklaring af hvordan de implementeres ved hjælp af Matlab.

#### BRUTE FORCE

Ved *Brute force* dannes et MATLAB script, hvor der genereres parametersæt for alle de mulige løsninger. Hvis der eksempelvis er to parametre, som kan ændres fra 0,0 til 1,0 med en interval på 0,1, vil programmet genere 100 forskellige parametersæt, som den kører tilfældigt igennem. Hver gang en simulering er færdig vil programmet gemme parametersættet og scoren, som beskriver, hvor stor forskel der er mellem rejsetiden i VISSIM og DataFromSky.

Fordelen ved denne metoder er, at det er relativt nemt at sætte den op. Der bliver genereret et antal parametersæt, svarende til alle mulige kombinationer. Metoden tager ikke højde for, hvor gode de andre parametersæt var og kører blot alle igennem, hvilket ikke er optimalt. Samtidig stiger beregningstiden markant ved flere parametre, som vist i tabel 2-1.



Figur 2-8 Fremgangsmåde for kalibrering med *Brute force*.

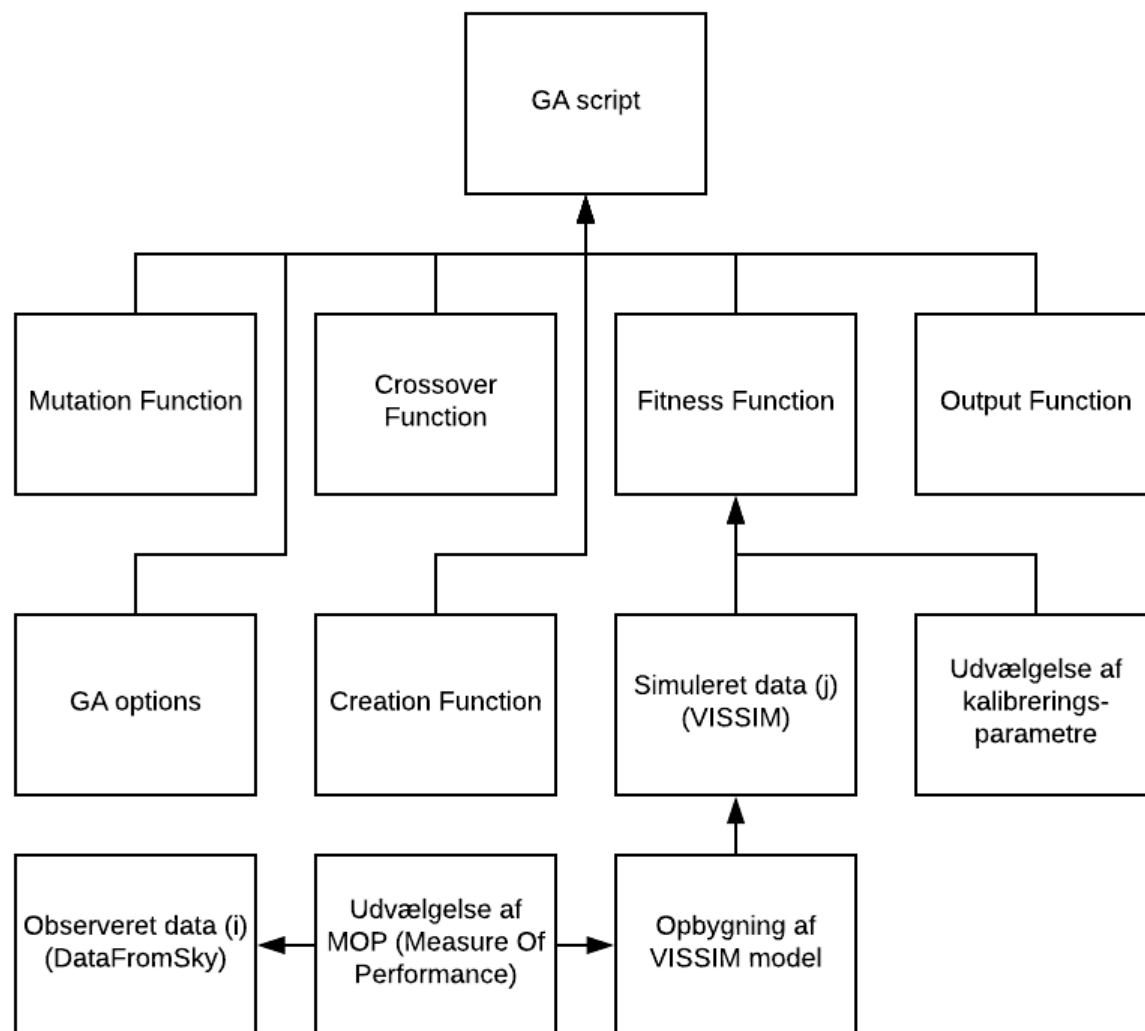
I praksis indsamles data fra DataFromSky i forhold til den udvalgte MOP (Measure Of Performance), som i dette tilfælde er rejsetid for hver rute i hvert minut, samtidig lægges der dataopsamlingsnit ind i VISSIM modellen, som kan måle på rejsetid under de samme forudsætninger, som DataFromSky. Herefter vælges et antal simuleringer, hvor der generes et antal parametersæt til de valgte kalibreringsparametre. I dette tilfælde parametre til *Driving Behavior* og *Conflict Areas*. Disse parametersæt bliver genereret helt tilfældigt og herefter indsættes de i VISSIM og simuleringen

starter. Forskellen mellem DataFromSky og VISSIM regnes som summen af rejsetid for hele systemet og betegnes som *Score*. Hvis scoren er bedre end den tidligere bedste score, så gemmes den sammen med parametersættet, som den nye bedste score. Denne proces fortsætter indtil alle de valgte simuleringer er færdiggjort. Da hver simulering tager ca. 2,5 minut sættes antallet af valgte simuleringer i dette projekt til 400, hvilket svarer til ca. 16 timer. Simuleringstiden på 16 timer er valgt, da det muliggør en kalibrering fra kl. 16 om

eftermiddagen til kl. 8 om morgenen dagen efter. For at kunne se betydningen af de enkelte runs og parametre gemmes resultaterne for hver simulering med det dertilhørende parametersæt. Selve Matlab scriptet til Brute force, kan ses i det elektroniske bilag.

Metoden kan forenkles ved enten at sætte intervallerne op eller ved at køre et bestemt antal runs. Det medfører dog, at ikke alle kombinationer bliver fundet og da algoritmen vælger tilfældige værdier er der en chance for, at den bedste kombination af parametre ikke findes i forhold til det valgte antal runs.





Figur 2-9 Opbygning af optimering med Genetisk Algoritme i MATLAB.

| PARAMETRE | SIMULERINGER |
|-----------|--------------|
| 2         | 100          |
| 3         | 1.000        |
| 4         | 10.000       |
| 5         | 100.000      |

Tabel 2-1 Antal simuleringer i forhold til antal parametre, hvis hver parameter har 10 mulige værdier og alle muligheder parameter-sæt skal benyttes.

#### GENETISK ALGORITME

Hvor *Brute force* ikke tager højde for de tidligere simuleringer, vil metoden med en genetisk algoritme evaluere på de bedste forsøg og danne nye forsøg herefter, som beskrevet i kapitel 1.

I MATLAB opbygges et script, som vist på figur 2-9, der benytter den indbyggede funktion til genetiske algoritmer med tilhørende funktioner. Som udgangspunkt anvender GA funktionen en række standard værdier og indstillinger i forhold til opsætning, men da dette er et relativt kompleks problem, skrives en del af koden selv efter behov. I det følgende vil de udvalgte indstillinger og

værdier til GA funktionen blive forklaret og fremgangsmåden for opsætning af disse scripts vil blive gennemgået. Der vil i det følgende blive præsenteret en del fagtermer. En forklaring af disse i forhold til dette projekt er i ordlisten i starten af rapporten.

### GENERELLE INDSTILLINGER

---

Ligesom for Brute force arbejdes der hen mod en simuleringstid på omtrent 16 timer, da det omtrent svarer til den tid, der er mellem en almindelig rådgiver går hjem fra arbejde til personen møder igen. For at imødekomme dette indstilles den genetiske algoritme således, at simuleringstiden svarer til de 16 timer. Da hver simulering i VISSIM tager omtrent 2,5 minut indstilles *populationsize* og generation således at de 16 timer opnås. For at algoritmen kan danne mange forsøg i starten sættes *populationsize* til 50 med 8 generationer. Det vil sige, at der først generes 50 tilfældige individuals (parametersæt), som får lov at køre over 8 generationer. Antallet af EliteParents/EliteChildren, individuals som fø-

res direkte til næste generation, indtastes også som en af de generelle indstillinger, i dette tilfælde 3 EliteChildren/EliteParents for hver generation.

### CREATION FUNCTION

---

Den første funktion, *Creation Function*, danner det umiddelbart første gæt, initial population, som benyttes i den genetiske algoritme. Til denne benyttes den tidligere valgte *populationsize*, som definerer hvor mange individuals, parametersæt, algoritmen skal teste for hver generation. Der laves derfor en funktion, der genererer en matrice, som er afhængig af antallet af individuals og antallet af variabler med tilfældige tal. I dette tilfælde en 50 x 5 matrice med tilfældige værdier inden for det givne interval for hver parameter. Outputtet af denne funktion er derfor en matrice med den initierende population.

VISSIM opererer med en præcision på én decimal, hvilket er nødvendigt at tage højde for i forhold til den enkelte individual. Derfor inkorporeres en række delfunktioner i *Creation Function*, som sik-

rer, at de tilfældige tal kun har én decimal. Da disse individuals er dannet tilfældigt, er der en risiko for, at der bliver generet parametersæt med fuldstændig samme værdier. Derfor søges der gennem matricen for unikke værdier, og hvis den finder værdier, som er ens, indsætter den selv andre værdier, således alle er unikke.

### FITNESS FUNCTION

---

*Fitness funktionen* beskriver hele optimeringsproblemet, og skal fodre algoritmen med resultater om, hvor god hver enkelt individual har været. For et projekt som dette, er denne funktion den vigtigste, da den evaluere hver enkelt parametersæt, individual, og videregiver informationer om, hvor gode parametersættene har været til algoritmen.

Da målet er at kalibrere modellen i forhold til virkeligheden, forsøger algoritmen at minimere forskellen mellem virkelighed (DataFromSky) og model (VISSIM). Dette gøres ved at ændre de fem parametre, tre til *Driving Behavior* og to til *Conflict Areas* fra VISSIM. Der opsættes derfor en fitness funktion, som kører

en simulering igennem med et bestemt parametersæt for derefter at beregne en *score*. *Scoren* er forskellen i rejsetid mellem DataFromSky og VISSIM. Der regnes for hvert parametersæt fra Creation Function en *score*, som benyttes videre i processen. Målet er derfor at arbejde hen imod en så lav score som muligt, da forskellen mellem DataFromSky og VISSIM vil være minimeret.

Forskellen mellem DataFromSky og VISSIM kvadreres for at undgå situationer, hvor positive og negative forskelle udligner hinanden. På den måde medtages både negative og positive forskelle mellem DataFromSky og VISSIM.

#### FITNESS SCALING FUNCTION

Baseret på tidligere benævnte *Score* fra Fitness funktionen skales alle Scores i forhold til hinanden, så der er større sandsynlighed for at gå videre med et parametersæt, individual, hvor den pågældende *score* er høj. Her benyttes MATLABs egen funktion, da den passer til denne optimeringstypes behov.

#### SELECTION FUNCTION

Selection funktionen finder de individuals, som skal benyttes i den næste generation, kaldet *parents*. Baseret på de skalerede værdier danner funktionen en linje, hvor alle *parents* indgår. Hver *parent* er afspejlet som et linjestykke, hvor længden af linjestykket er afhængig af tidligere beskrevne Fitness Scaling Function. På den måde er der større chance for, at de individer med bedst score bliver brugt videre til næste generation.

#### CROSSOVER FUNCTION

Efter *Parents* er valgt ved hjælp af *Selection Function* findes antallet af *Children* dannet ved Crossover. I dette projekt benyttes en *CrossoverFraction* på 0,8, hvilket angiver, hvor mange *Children* der skal dannes ved hjælp af *Crossover*. I dette tilfælde, hvor populationsize er 50 vil 40 *Children* blive dannet ved hjælp af Crossover. Selection Functionen henter herefter 40 *Parents* par, som krydses for at danne 40 *Children*.

Der findes flere forskellige typer af Crossover alt efter hvor mange gange der

skal krydses. Der opbygges for denne algoritme en funktion, som krydser ét sted. Dette gøres ved at finde et tilfældigt tal mellem 1 og antallet af variabler, hvor den splitter to *parents* op i forhold til den tilfældige værdi. Et eksempel er vist i tabel 2-2, hvor der foretages crossover efter 2. variabel.

|          |   |   |   |   |   |
|----------|---|---|---|---|---|
| Parent1  | A | B | C | D | E |
| Parent2  | 1 | 2 | 3 | 4 | 5 |
| Children | A | B | 3 | 4 | 5 |

Tabel 2-2 Eksempel på crossover.

#### MUTATION FUNCTION

De resterende *children*, som hverken er *EliteChildren* eller *CrossoverChildren*, dannes ved mutation. Mutationen foregår ved at en *parent* udvælges og herefter er der en vis sandsynlighed for, at en variabel ændres. I dette projekt er den sandsynlighed sat til 10%. Dette giver en stor sandsynlighed for, at der sker en mutation, og dermed kan algoritmen søge igennem flere parametre hurtigere. Hvis den bliver valgt, så muteres (ændres) parameteren i henhold til randbetingelserne, således at de ikke

overskrider det valgte interval for parameteren. Det muterede individual bruges herefter i næste generation, som et nyt children. Et eksempel på mutation, er vist i tabel 2-3.

|          |      |      |      |      |
|----------|------|------|------|------|
| Parent1  | 1    | 2    | 3    | 4    |
| Mutation | 0,67 | 0,32 | 0,08 | 0,89 |
| Children | 1    | 2    | 1    | 4    |

**Tabel 2-3 Eksempel på mutation.**

For GA algoritmen er der flere forskellige metoder til at fremtvinge denne mutation. Fælles for alle de prædefinerede mutationsfunktioner er, at de muterer på langt flere decimaler end nødvendigt, da VISSIM kun bruger én decimal. Grunden til at antal decimaler i mutationen er vigtig er, at hvis MATLAB muterer fra fx 0,5487 til 0,5418, så vil VISSIM læse 0,6 uanset hvad. Dermed antager MATLAB at den har lavet en mutation, altså ændret på et individual, men i praksis, er der ikke sket nogen forskel.

## OUTPUT FUNCTION

Formålet med en output function er, at få lagret information for hver simulering. MATLABs indbyggede GA funktion kan give resultater for den sidste generation, men for at kunne se, hvordan algoritmen forbedrer sig for hver generation skrives en output function, som kan lagre informationer om score og individuals for hver simulering og generation.

Output funktionen gemmer derfor informationer for hver simulering om den pågældende simuleringsscore og parametersæt. Output funktionen sørger også for, at scores og parametersæt bliver inddelt efter generationer, så de kan holdes op imod hinanden bagefter og processen kan illustreres.

Alle MATLAB scripts til GA, kan ses i det elektroniske bilag.

## 2.5.3 EVALUERING

Fælles for begge simuleringer er måden de evalueres på. I DataFromSky beregnes rejsetiden for alle mulige ruter for hvert minut i spidstimen, og den rejsetid

holdes op mod et lignende rejsetid fra VISSIM, som også beregnes for alle mulige ruter for hvert minut i spidstimen. Alle rejsetider, både fra DataFromSky og VISSIM, summeres op og den totale forskel mellem summen af DataFromSky og VISSIM kan herefter beregnes.

Den samlede forskel mellem VISSIM og DataFromSky kaldes i dette projekt for simuleringsscore. Formålet for optimeringsalgoritmerne såsom GA er derfor at minimere denne score, så den absolutte forskel skal være så tæt på nul som muligt. Denne score kan MATLAB beregne for hver simulering, og det antages derfor at det parametersæt, som giver den bedste score, må være den bedste kalibrering.

Der evalueres kun hele systemet, da algoritmerne skal have et samlet tal at evaluere på. Det betyder, at der kan være urealistisk meget kø i visse spor i simuleringen, som ikke kan ses i virkeligheden. Denne problemstilling er yderligere forklaret i diskussionsafsnittet i kapitel 5.

---

Med de "optimale" kalibreringer fra Brute force og GA udtrækkes data fra VISSIM om rejsetid for hver rute for hvert minut og de forskellige kalibreringer sammenholdes med den anbefalede kalibrering fra Vejreglen for mikrosimulering og DataFromSky.

De forskellige resultater sammenholdes for at se, hvor god VISSIM er til at modellere virkeligheden, samt hvor god den automatiserede kalibrering er, og dermed om det overhovedet kan svare sig at benytte tilføje denne automatiserede proces.

Udover at evaluere på denne spidstime-model forsøges samtidig også med de mest kritiske 30 minutter for at se, om VISSIM foreslår en anden kalibrering, og om de kan modellere fremkommelighedsproblemerne bedre. Som udgangspunkt udføres samme model med den lille ændring, at der kun skal modelleres fra 7:30 til 8:00, hvorfor det bedste parametersæt sandsynligvis ændres.



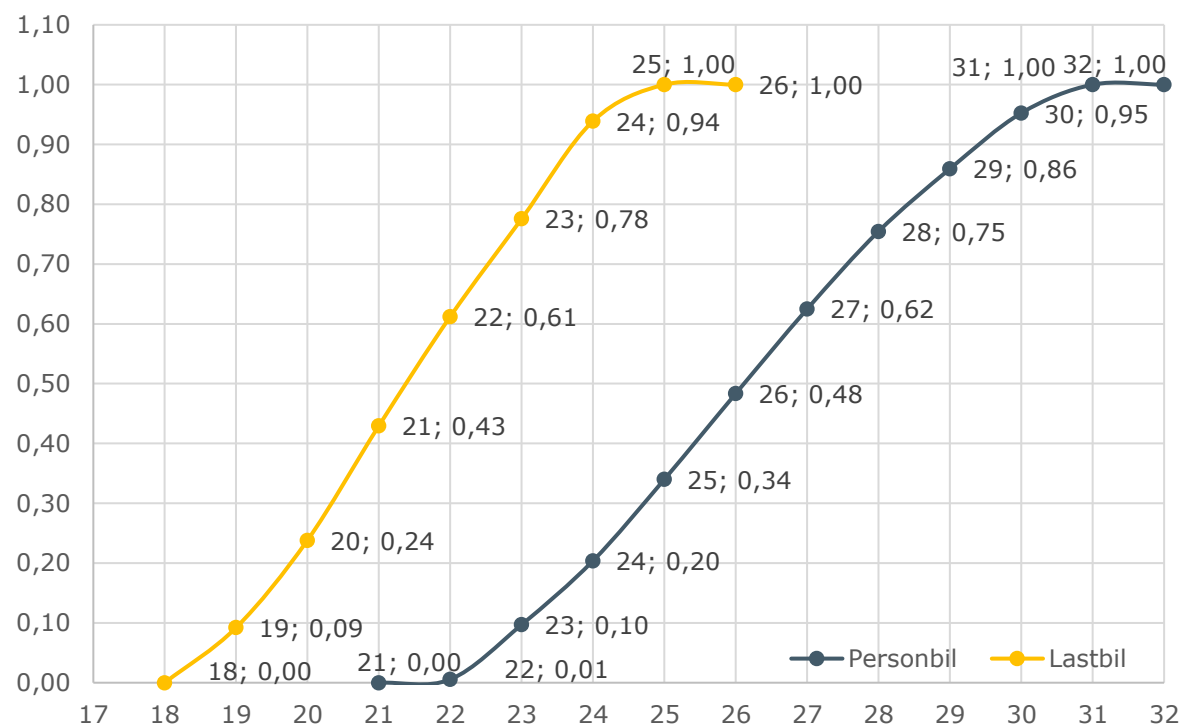


### 3 RESULTATER

I indeværende kapitel præsenteres resultaterne af projektets analyser. Først gennemgås resultaterne i forbindelse med opbygningen af VISSIM modellen, hvor DataFromSky bidrager med kalibreringsdata. Efterfølgende gennemgås fokuspunkter til den automatiserede kalibrering, og resultaterne for de forskellige optimeringsmetoder præsenteres. Til sidst sammenlignes resultaterne for de forskellige optimeringsmetoder med simuleringer, hvor vejreglernes værdier indgår og DataFromSky.

#### 3.1 OPBYGNING AF VISSIM MODEL

I opbygningen af en VISSIM model benyttes normalt standardværdier fra vejreglen og information fra observationer i marken. Med DataFromSky kan disse standardværdier erstattes med målte værdier direkte fra DataFromSky. Det muliggør en opbygning af VISSIM modellen, hvor der anvendes virkelige og stedspecifikke data til kalibreringen. I de



Figur 3-1 Kumuleret frekvenskurve for personbiler og lastbiler af deres gennemsnitlige hastigheder.

følgende underafsnit vil resultaterne for DataFromSky udtræk blive gennemgået.

#### 3.1.1 CIRKULATIONSHASTIGHED

Som redegjort for i kapitel 2, opsættes en metode til, at indsamle cirkulationshastigheder for rundkørslen. Hastigheden i cirkulationsarealet indgår til kali-

breringen af modellen, som en kendt variabel. I flere artikler fra litteraturstudiet anvendes hastighedsfordelinger, som variable parameter, der forsøges at optimere for, men da det med DataFromSky er muligt at trække virkelige data ud og benytte i VISSIM, foretrækkes dette. Resultatet af cirkulationsha-

stighedsanalysen viser, at gennemsnitshastigheden for personbiler og lastbiler i cirkulationsarealet i rundkørslen er henholdsvis 27,3 km/t og 22,5 km/t. Da DataFromSky har information fra hver enkelt trafikant kan der på baggrund af hastighedsudtrækket laves frekvenskurver. Da VISSIM er et mikrosimuleringsprogram, der dermed modellerer hvert enkelt køretøj individuelt, er der mulighed for at inkorporere netop frekvenskurven, således at trafikanterne har forskellige hastigheder, men gennemsnittet stadig ligger på 27,3 km/t og 22,5 km/t for henholdsvis personbiler og lastbiler. På figur 3-1 er frekvenskurverne, som indlægges i VISSIM for Reduced Speed Areas, indtegnet.

Effekten af den ændrede cirkulationshastighed forventes at have en relativ stor betydning for kalibreringen af modellen. Det skyldes, at VISSIM normalt blot benytte en lineær kurve fra 25 til 30 km/t, hvilket DataFromSkys frekvenskurve falsificerer.

### 3.1.2 OD MATRICE

I DataFromSky udtrækkes trajectories for hver bil i forhold til indlagte entry- og exit gates. På den måde opsamles data for alle ruter både i forhold til svingmønstre og belastning fra hver vejgren. Af tabel 3-1 fremgår OD matricen, som er dannet på baggrund af DataFromSky data. Det fremgår her, at den primære trafik kommer fra motorvejen mod Skærbækvej nord. Generelt skal størstedelen af trafikken mod Skærbækvej nord, hvor der er erhvervsområde samt motorvejstilkørsel i nordlig retning.

Svingmønstrene indtastes direkte i VISSIM under "Vehicle routing" på kvartersniveau, hvorefter VISSIM selv danner ankomstfordelingen baseret på de forskellige ruter og random seed. I VISSIM indtastes den totale mængde af trafik fra hvert ben under "Vehicle Input". Selvom trafikken er opdelt på kvartersniveau, skal inputtet til VISSIM indtastes som køretøjer pr. time, hvilket betyder, at alle input markeret med fed i tabel 3-1, skal opregnes fra kvartersniveau til

timeniveau. Det indledende kvarter benyttes til at fylde modellen op, således den ikke er helt tom fra kl. 7:00, hvor simuleringen starter. Da der ikke er data fra 6:45 til 7:00 indsættes 80% af trafikken fra 7:00 til 7:15.

| <b>Antal</b>                     | <b>07:00 - 07:15*</b> |            | <b>07:15 - 07:30</b> |           | <b>07:30 - 07:45</b> |           | <b>07:45 - 8.00</b> |           | <b>08:00 - 8.15</b> |           |
|----------------------------------|-----------------------|------------|----------------------|-----------|----------------------|-----------|---------------------|-----------|---------------------|-----------|
| <b>Fra &gt; Til</b>              | <b>PB*</b>            | <b>LB*</b> | <b>PB</b>            | <b>LB</b> | <b>PB</b>            | <b>LB</b> | <b>PB</b>           | <b>LB</b> | <b>PB</b>           | <b>LB</b> |
| <b>Total fra Adelvej SV</b>      | <b>67</b>             | <b>5</b>   | <b>92</b>            | <b>9</b>  | <b>65</b>            | <b>4</b>  | <b>50</b>           | <b>3</b>  | <b>66</b>           | <b>3</b>  |
| Adelvej SV > Motorvej            | 12                    | 1          | 22                   | 1         | 19                   | 1         | 16                  | 1         | 18                  | 0         |
| Adelvej SV > Skærbækvej nord     | 51                    | 4          | 66                   | 6         | 38                   | 2         | 29                  | 2         | 44                  | 3         |
| Adelvej SV > Skærbækvej syd      | 4                     | 0          | 4                    | 2         | 7                    | 1         | 5                   | 0         | 4                   | 0         |
| <b>Total fra Motorvej</b>        | <b>122</b>            | <b>11</b>  | <b>161</b>           | <b>16</b> | <b>111</b>           | <b>16</b> | <b>147</b>          | <b>11</b> | <b>118</b>          | <b>14</b> |
| Motorvej > Adelvej SV            | 9                     | 2          | 14                   | 3         | 18                   | 2         | 19                  | 2         | 11                  | 2         |
| Motorvej > Skærbækvej nord       | 81                    | 6          | 118                  | 8         | 46                   | 7         | 80                  | 4         | 82                  | 7         |
| Motorvej > Skærbækvej syd        | 32                    | 3          | 29                   | 5         | 47                   | 7         | 48                  | 5         | 25                  | 5         |
| <b>Total fra Skærbækvej nord</b> | <b>64</b>             | <b>7</b>   | <b>72</b>            | <b>3</b>  | <b>102</b>           | <b>8</b>  | <b>104</b>          | <b>5</b>  | <b>84</b>           | <b>6</b>  |
| Skærbækvej nord > Adelvej SV     | 21                    | 0          | 20                   | 1         | 31                   | 3         | 37                  | 1         | 24                  | 1         |
| Skærbækvej nord > Motorvej       | 8                     | 3          | 4                    | 1         | 10                   | 0         | 2                   | 1         | 4                   | 1         |
| Skærbækvej nord > Skærbækvej syd | 34                    | 4          | 47                   | 1         | 60                   | 5         | 63                  | 3         | 56                  | 4         |
| <b>Total fra Skærbækvej syd</b>  | <b>78</b>             | <b>11</b>  | <b>83</b>            | <b>7</b>  | <b>41</b>            | <b>8</b>  | <b>66</b>           | <b>15</b> | <b>66</b>           | <b>11</b> |
| Skærbækvej syd > Adelvej SV      | 2                     | 0          | 4                    | 0         | 2                    | 1         | 3                   | 0         | 6                   | 0         |
| Skærbækvej syd > Motorvej        | 19                    | 3          | 11                   | 2         | 13                   | 3         | 19                  | 3         | 9                   | 0         |
| Skærbækvej syd > Skærbækvej nord | 57                    | 8          | 68                   | 5         | 26                   | 3         | 44                  | 12        | 51                  | 11        |

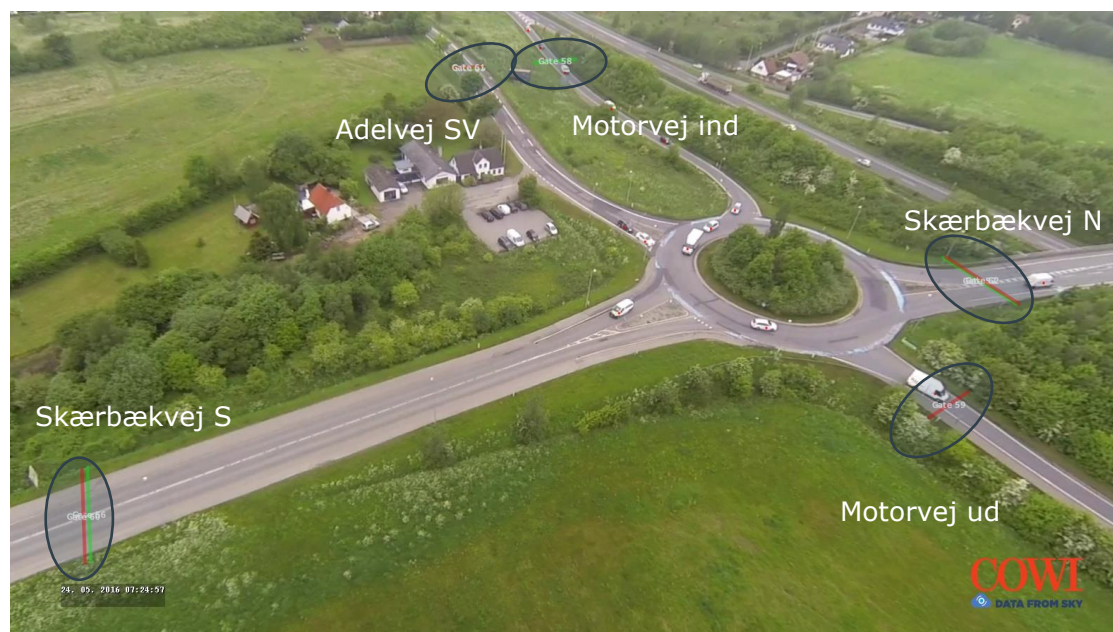
Tabel 3-1 OD Matrice for spidstimen samt indledende kvarter. Det indledende kvarter er markeret med \*. PB = Personbiler. LB = Lastbiler.

| Tidsinterval  | MV - N      | MV - S      | MV - SV     | N - MV      | N - S       | N - SV      | S - MV      | S - N       | S - SV      | SV - MV     | SV - N      | SV - S      |
|---------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| 07:15 - 07:20 | 27,5        | 26,7        | 29,8        | 17,4        | 21,7        | 18,8        | 26,1        | 25,9        | 35,9        | 31,4        | 39,4        | 51,9        |
| 07:20 - 07:25 | 28,1        | 27,8        | 31,5        | 23,2        | 19,9        | 18,8        | 22,8        | 29,0        | 35,9        | 28,2        | 28,3        | 38,7        |
| 07:25 - 07:30 | 32,7        | 29,1        | 29,1        | 23,1        | 22,4        | 19,4        | 38,6        | 38,5        | 41,6        | 36,3        | 46,0        | 56,3        |
| 07:30 - 07:35 | 28,1        | 29,3        | 32,1        | 18,4        | 20,8        | 18,3        | 26,0        | 26,3        | 51,4        | 31,9        | 32,0        | 46,0        |
| 07:35 - 07:40 | 39,0        | 41,3        | 47,5        | 17,5        | 21,4        | 19,0        | 29,2        | 29,1        | 42,9        | 49,1        | 46,8        | 46,1        |
| 07:40 - 07:45 | 54,5        | 52,1        | 58,1        | 19,3        | 23,6        | 21,2        | 84,1        | 85,0        | 67,8        | 68,7        | 74,3        | 71,6        |
| 07:45 - 07:50 | 52,1        | 54,1        | 50,7        | 19,7        | 21,5        | 20,0        | 83,3        | 88,7        | 80,6        | 66,2        | 62,0        | 87,6        |
| 07:50 - 07:55 | 64,1        | 64,1        | 64,3        | 23,3        | 22,9        | 23,7        | 73,0        | 79,6        | 75,2        | 48,7        | 69,0        | 63,4        |
| 07:55 - 08:00 | 47,5        | 47,2        | 42,2        | 25,4        | 19,8        | 20,1        | 70,5        | 55,6        | 54,5        | 59,1        | 60,7        | 73,5        |
| 08:00 - 08:15 | 32,7        | 32,2        | 24,9        | 24,7        | 19,4        | 18,5        | 33,2        | 30,1        | 45,1        | 37,3        | 33,9        | 53,0        |
| 08:05 - 08:10 | 29,3        | 25,6        | 32,6        | 25,1        | 21,7        | 19,2        | 22,7        | 26,6        | 37,6        | 30,9        | 31,3        | 35,1        |
| 08:10 - 08:15 | 29,1        | 26,5        | 25,1        | 15,4        | 20,8        | 18,2        | 16,1        | 24,2        | 35,5        | 19,4        | 33,4        | 19,0        |
| <b>GNS.</b>   | <b>38,7</b> | <b>38,0</b> | <b>39,0</b> | <b>21,0</b> | <b>21,3</b> | <b>19,6</b> | <b>43,8</b> | <b>44,9</b> | <b>50,3</b> | <b>42,3</b> | <b>46,4</b> | <b>53,5</b> |

Tabel 3-2 Rejsetidsmatrice opdelt i 5 minutters intervaller. MV = Motorvej. SV = Adelvej SV. S = Skærbækvej S. N = Skærbækvej N.

### 3.1.3 REJSETID

I DataFromSky indlægges snit, som vist på figur 3-2, hvor rejsetiden for de forskellige ruter for hvert minut er trukket ud. Den fulde rejsetidsmatrice er vedlagt i bilag B, men for overblikket skyld, er rejsetiden inddelt i 5 minutters intervaller, som vist i tabel 3-2. Her fremgår det, at jo længere vi kommer ind i spidstimen, des længere bliver rejsetiden for de fleste ben, da rundkørslen dermed bliver mere belastet.



Figur 3-2 Indlagte snit i DataFromSky til indsamling af rejsetid.

Som beskrevet i kapitel 2, er der for visse minuttal ikke registreret nogen rejsetid, hvorfor der er interpoleret, for at få omtrentlige værdier. De 5 minutters intervaller, hvor der indgår interpolerede værdier fra minimum et minut, er markeret med rød i tabel 3-2.

I bilag B er vedlagt en tabel med rejsetiden pr. minut samt rejsetidsdiagrammer for de enkelte ben, som viser, rejsetiden for de forskellige ruter i forhold til tiden.

### 3.2 OPTIMERINGSALGORITMER

*I følgende afsnit vil resultaterne for de i projektet valgte optimeringsprocesser blive vist, hvor også valg foretaget undervejs og resultater vil blive gransket. De forskellige algoritmer vil blive gennemgået individuelt, hvorefter der til sidst samles op og vurderes på løsningerne i forhold til hele optimeringsproblemet samlet.*

#### 3.2.1 GENERELLE FORUDSÆTNINGER FOR OPTIMERINGSALGORITMERNE

Da disse algoritmer kan køre i flere uger uden at have et endegyldigt svar, har det været nødvendigt at begrænse simuleringstiden. Her er valgt 16 timer, da det svarer til tiden fra en almindelig kontoransat går hjem til personen møder igen. Dette har selvfølgelig nogle begrænsninger i forhold til at finde den optimale kalibrering.

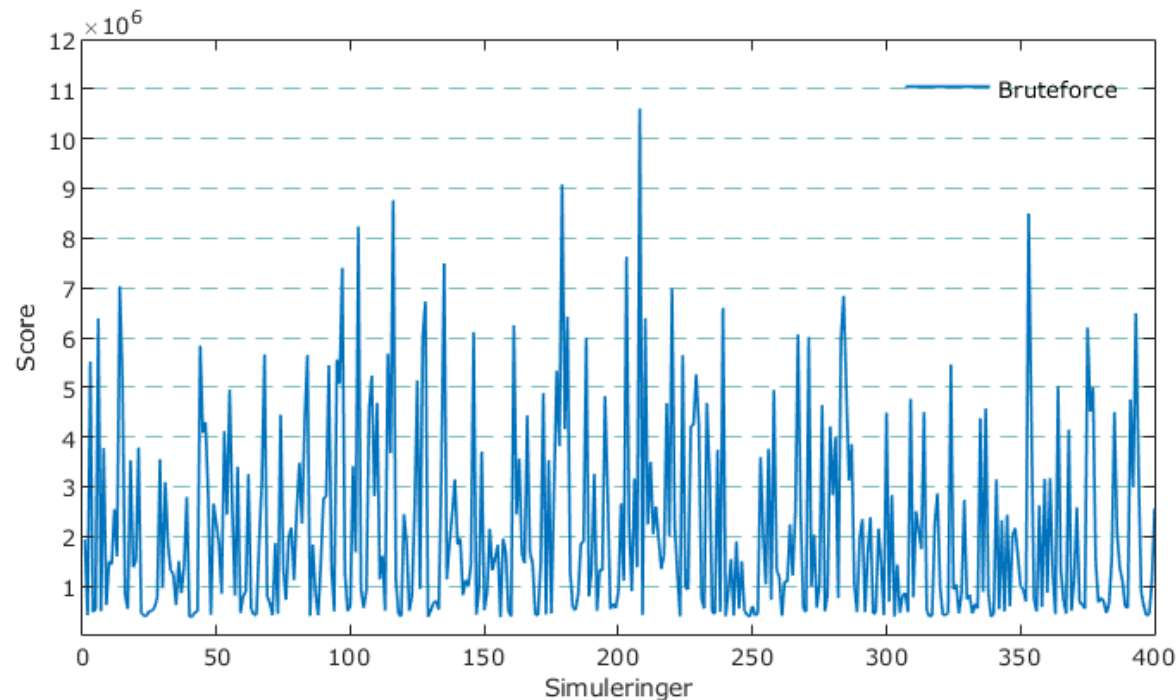
For at give algoritmerne mulighed for at køre flere simuleringer igennem på de 16 timer er antallet af *runs* i VISSIM blevet sat ned til 5. Det anbefalede antal fra COWI og vejreglen er 25 runs per simulering, hvilket ville gøre den relative lange simuleringstid 5 gange længere. Det betyder, at dårlige ankomstfordelinger vil have større betydning. Da der i VISSIM benyttes samme *random seeds* for de forskellige runs, burde det ikke have den store betydning, da de derfor har samme forudsætninger for den samlede score.

Ud fra litteraturstudiet og erfaringer skabt i mit tidligere projekt, er 5 variabler valgt ud, som ændres for begge algoritmer gennem hver simulering. Det omhandler tre variabler til *Driving Behavior* og to til *Conflict Areas*. For *Conflict Areas* er der justeret på *Front Gap* og *Safety Distance factor*, som kan variere mellem 0,1 og 1,0 med et interval på 1,0. For *Driving Behavior* er der justeret på  $ax$ ,  $bx_{mult}$  og  $bx_{add}$ , hvor  $ax$  og  $bx_{add}$  kan variere mellem 1,0 og 3,0 samt  $bx_{mult}$  kan variere mellem 2,0 og 4,0 alt sammen med et interval på 0,1. Disse randbetingelser er dannet på baggrund af litteraturstudiet, erfaring, diskussioner med fagpersoner hos COWI og anbefalinger i vejreglen/VISSIM manualen.

Konsekvensen af forudsætninger som de ovenstående er alle diskuteret i kapitel 4.

#### 3.2.2 BRUTE FORCE

Brute force algoritmen genererer 400 tilfældige parametersæt, som alle sammen er indenfor randbetingelserne. På



Figur 3-3 Resultatet af Brute force algoritmen for 400 simuleringer.

figur 3-3 er resultatet af Brute force algoritmen vist, hvor scoren er plottet som en funktion af simuleringnummer. Her fremgår det, at der ikke er nogen systematik i søgningen efter det bedste parametersæt. Efter hver simulering er parametersættet og den tilhørende score gemt. Af figur 3-3 fremgår det, at Brute force relativt hurtigt finder nogle lave

værdier, men der er ingen garanti for, hvor hurtigt algoritmen finder dem.

Ydermere viser de forskellige scores, at modellen skal kalibreres med relativt lave værdier for at opnå den bedste kalibrering. Ud fra de indsamlede bedste scores fremgår det, at de bedste scores altid bliver erstattet af et parametersæt med lavere værdier, hvilket fremgår af

tabel 3-3. Ret hurtigt finder Brute force frem til, at Safety Distance Factor skal være relativt lav og samtidig viser resultaterne også, at parametrene til Driving Behavior bliver lavere i takt med at scoren falder.

| FG         | SF         | AX         | BX <sub>add</sub> | BX <sub>mult</sub> | Sim.       |
|------------|------------|------------|-------------------|--------------------|------------|
| 0,9        | 0,5        | 2,4        | 2,0               | 3,4                | 1          |
| 1,0        | 0,1        | 1,2        | 2,0               | 2,9                | 2          |
| 0,9        | 0,1        | 1,4        | 1,5               | 3,2                | 23         |
| 1,0        | 0,3        | 1,1        | 1,5               | 2,2                | 24         |
| 0,1        | 0,1        | 1,2        | 1,1               | 2,9                | 40         |
| 0,5        | 0,3        | 1,2        | 1,6               | 2,1                | 41         |
| 0,3        | 0,2        | 1,8        | 1,2               | 2,7                | 129        |
| <b>0,5</b> | <b>0,1</b> | <b>1,3</b> | <b>1,7</b>        | <b>2,4</b>         | <b>156</b> |

Tabel 3-3 Liste over de bedste parametersæt i løbet af de 400 simuleringer. FG = Front Gap. SF = Safety Distance Factor. Sim = Simuleringens nummer. Det bedste parametersæt er markeret med fed.

Alt kodning er udført i MATLAB, hvilket er vedlagt i det elektroniske bilag.

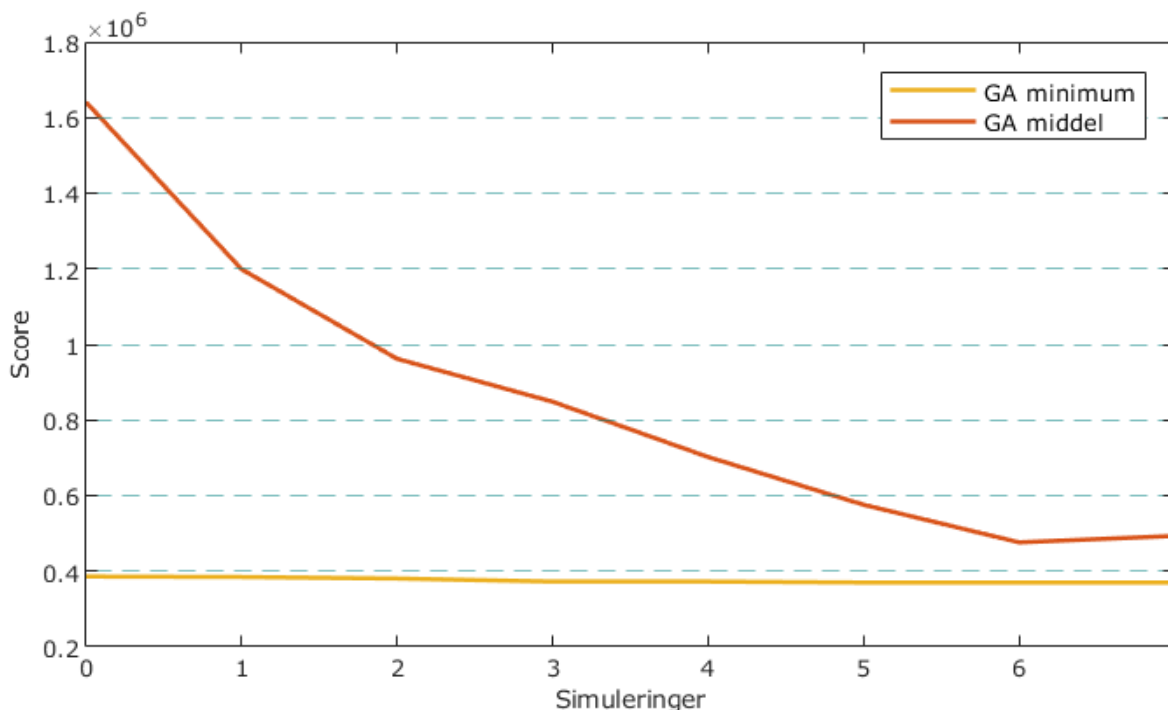


### 3.2.3 GENETISK ALGORITME

Den genetiske algoritme har 7 generationer samt en indledende generation, hvor hver generation har populationsstørrelse på 50. Dermed omfatter den genetisk algoritme også 400 simuleringer, hvilket er samme antal som Brute force.

Resultatet af den genetiske algoritme, kan ses på figur 3-4, hvor det fremgår, at algoritmen relativt hurtigt finder et lavpunkt, men alligevel finder mere optimale værdier, som kan justere en lille smule på det optimale parametersæt. Det fremgår også, hvordan de tre Driving Behavior værdier bliver mindre i takt med en udvikling i generationerne.

| FG         | SF         | AX         | BX <sub>add</sub> | BX <sub>mult</sub> | Gen.     |
|------------|------------|------------|-------------------|--------------------|----------|
| 0,5        | 0,2        | 1,1        | 1,6               | 3,5                | 0        |
| 0,4        | 0,3        | 1,8        | 1,2               | 2,4                | 1        |
| 0,8        | 0,1        | 1,2        | 1,4               | 2,6                | 2        |
| 0,5        | 0,2        | 1,1        | 1,0               | 2,3                | 3        |
| 0,5        | 0,2        | 1,1        | 1,0               | 2,3                | 4        |
| 0,3        | 0,4        | 1,1        | 1,0               | 2,3                | 5        |
| 0,3        | 0,4        | 1,1        | 1,0               | 2,3                | 6        |
| <b>0,3</b> | <b>0,4</b> | <b>1,1</b> | <b>1,0</b>        | <b>2,3</b>         | <b>7</b> |

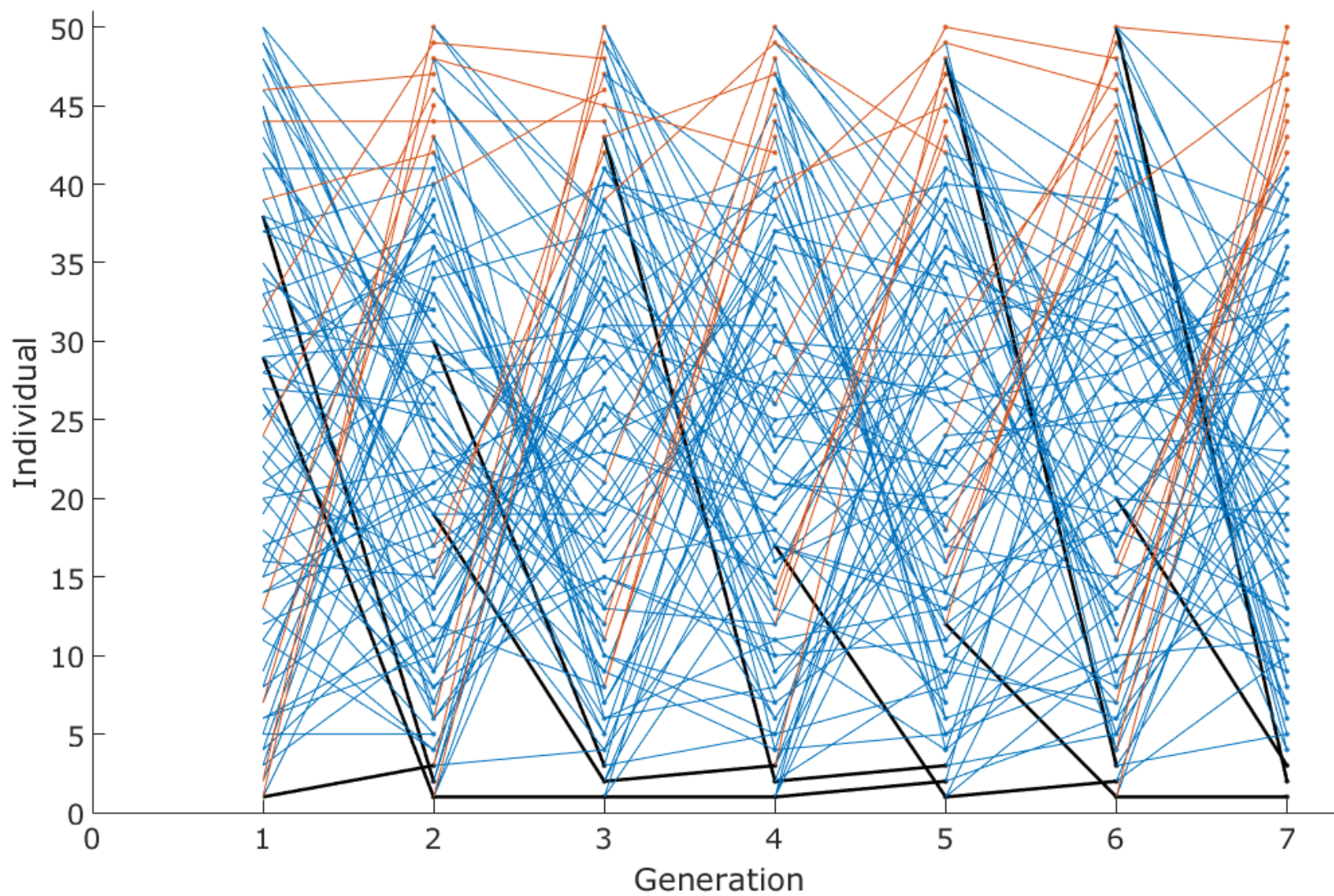


**Figur 3-4** Resultat af GA algoritmen, hvor både minimumsværdierne og middelværdierne er indtegnet.

**Tabel 3-4** Udvikling i bedste parametersæt for GA. FG = Front Gap. SF = Safety Distance Factor. Gen. = Generation. Det bedste parametersæt er markeret med fed.

De genetiske algoritme bruger hele tiden scoren til at udvælge, hvilke individuals (parametre), som skal videre til næste generation, hvor en god (lav) score gør chancen højere. Det betyder, at jo flere

generationer, som algoritmen når igennem, des større chance er der for at finde de lave tal. Det fremgår af tabel 3-4, hvordan gennemsnittet (GA middel) af de 50 scores falder i takt med generationerne, således den kommer meget tæt på den laveste parameter, som er angivet med orange.

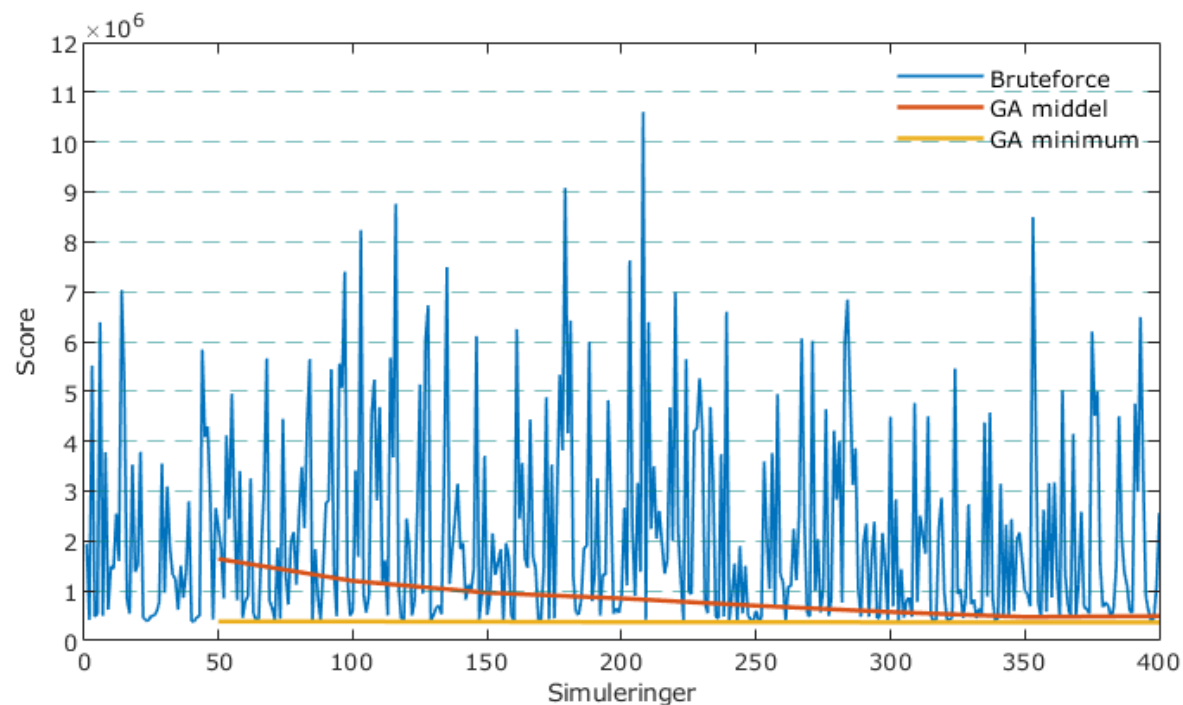


Figur 3-5 Genplot fra den genetisk algoritme. Sort er Elitechildren, Blå er Crossoverchildren og Rød er Mutationchildren.

I MATLAB er det desuden muligt at lagre informationer fra hver generation i forhold til hvilke parents, som bliver udvalgt og brugt til at skabe nye parents samt hvilke children, som er dannet som elitechildren, mutationchildren eller crossoverchildren. Af figur 3-5 fremgår hvilke parents, som går videre fra hver generation, og hvordan de forskellige children er skabt. Dette genplot viser, at den genetiske algoritme ikke er stagneret helt, da der stadig er stor udskiftning i EliteChildren, som er markeret med sort på figur 3-5. Det betyder, at der med fordel kunne været simuleret længere for at være mere sikker på, at det bedste parametersæt var fundet.

#### 3.2.4 SAMLET VURDERING AF OPTIMERINGSALGORITMERNE

Hvor begge optimeringsalgoritmer har deres fordele, er det klart, at den genetiske algoritme opererer på et helt andet niveau end Brute force. Det skyldes, blandt andet den kodning og proces, som foregår i forbindelse med GA algoritmen. Hvor Brute force blot afprøver



**Figur 3-6** Kombineret graf, som viser Brute force i forhold til GA.

tilfældige kombinationer, kan den genetiske algoritme frasortere de dårlige individer, således de ikke indgår i næste generation. På den måde, som figur 3-6 også viser, vil den genetiske algoritme være garant for at blive bedre i takt med at algoritmen får lov at arbejde indtil den konvergerer. Derimod vil Brute force blot være tilfældige nedslagspunkter i

parameteropsætningens udfaldsrum, hvor den første simulering kan være lige så god som den sidste simulering med mindre alle mulige parameteropsætninger simuleres, hvilket vil være utrolig tidskrævende med simuleringstider over 300 dage med det nuværende setup. I dette projekt er Brute force blot sat op til at generere 400 tilfældige værdier, og

de resultater, som er vist på figur 3-6, er blot tilfældig udvalgte parametre. Hvis MATLAB scriptet blev kørt igen, ville det generere 400 nye tilfældige parametersæt, som denne gang kunne enten være bedre eller dårligere.

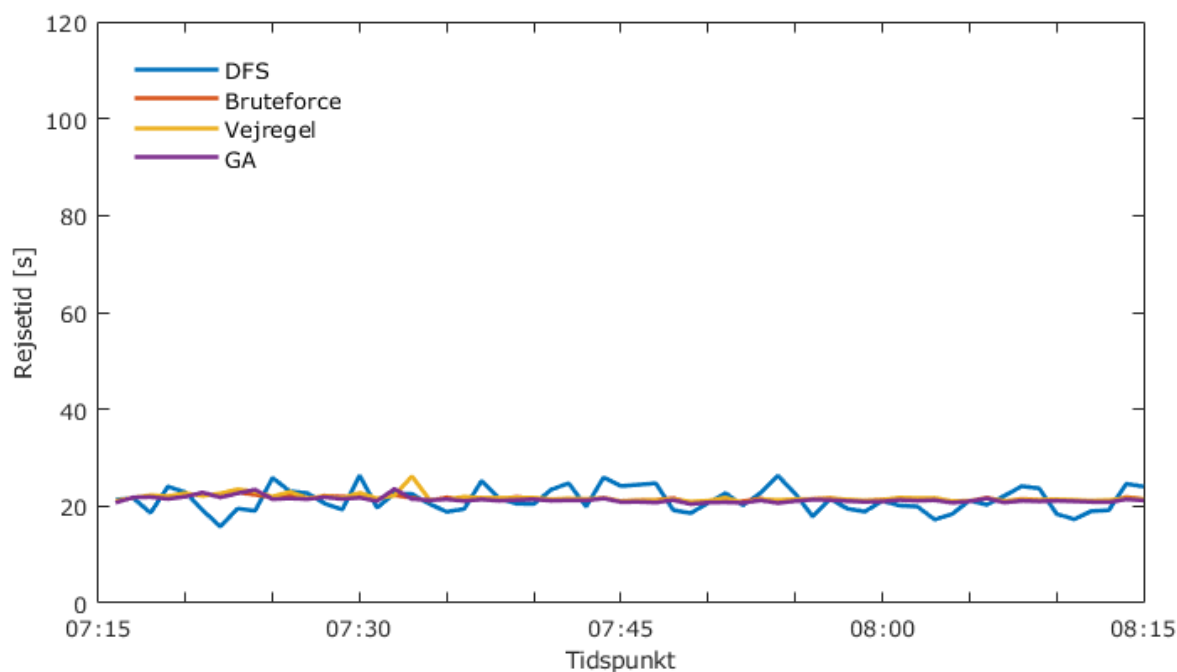
I dette projekt vurderes de forskellige simuleringer ved scoren, som beskriver hvor godt simuleringer passer i forhold til DataFromSky. Hvis scoren fra GA betragtes i forhold til scoren fra Brute force, fremgår det, at den genetisk algoritme systematisk optimerer på problemet, hvorimod Brute force er meget afhængig af den tilfældige generering af parametersæt.

### 3.3 SAMMENLIGNING AF DET BEDSTE PARAMETERSÆT

Med udgangspunkt i de bedste parametersæt fra de to optimeringsalgoritmer og standardparametre fra vejreglen, foretages nye simuleringer i VISSIM med henblik på at undersøge, hvor godt de passer i forhold til de oprindelige rejsetider fra DataFromSky. I VISSIM benyttes samme model, hvor det kun er de fem kalibreringsparametre, som er ændret, hvilket giver alle modellerne samme udgangspunkt. I følgende afsnit vil der kun blive præsenteret et par udvalgte repræsentative ruter, hvorimod de resterende vil finde sig i bilag C. I det bilag vil både rejsetidsmatricer og grafer for hver rute blive præsenteret.

Ud fra optimeringsalgoritmerne Brute Force og GA samt de anbefalede værdier fra vejreglen for mikrosimulering er følgende parametersæt benyttet i VISSIM:

| FG  | SF  | AX  | BX <sub>add</sub> | BX <sub>mult</sub> | Type |
|-----|-----|-----|-------------------|--------------------|------|
| 0,5 | 0,1 | 1,3 | 1,7               | 2,4                | BF   |
| 0,3 | 0,4 | 1,1 | 1,0               | 2,3                | GA   |
| 0,5 | 0,5 | 2,0 | 2,0               | 3,0                | VR   |



Figur 3-7 Rejsetid pr. minut fra Skærbækvej nord til Skærbækvej syd fra 07:15 til 08:15.

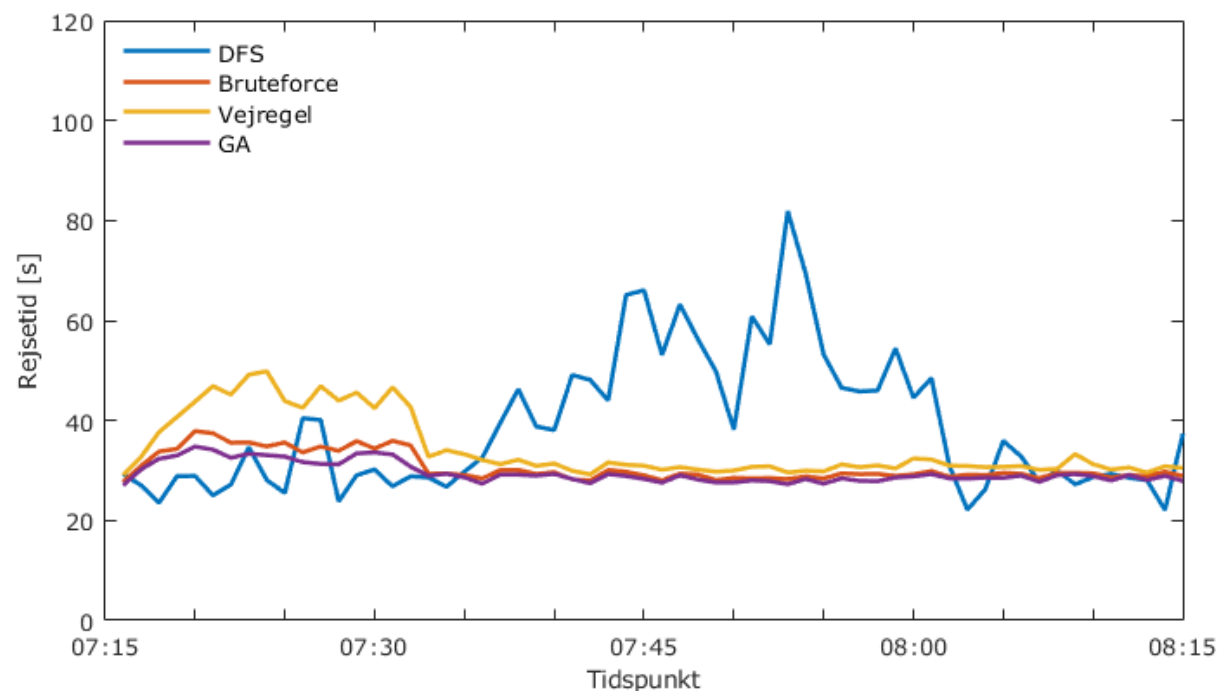
Tabel 3-5 Bedste rejsetider for henholdsvis Brute Force, GA og vejreglen. BF = Brute Force. GA = Genetisk Algoritme. VR = Vejreglen for mikrosimulering.

Af figur 3-7 fremgår det, hvordan rejsetiderne fra Skærbækvej nord til Skærbækvej syd for alle simuleringer stemmer nogenlunde overens med rejseti-

derne fra DataFromSky. Der er maksimalt en forskel på 5,5 sekunder mellem data fra DataFromSky og data fra de andre simuleringer. Generelt kan det siges, at VISSIM ikke har nogen problemer med at simulere ruter, hvor de pågældende ben ikke er særlig belastede og der dermed ikke er særlig mange fremkommelighedsproblemer. Skærbækvej

nord er ikke særlig belastet, da de fleste trafikanter kører fra ved Skærbækvej nord, hvilket gør, at de fleste trafikanter, som kommer fra Skærbækvej nord kan køre direkte ud i rundkørslen.

Hvis der derimod tages udgangspunkt i ruten fra motorvejen til netop Skærbækvej nord fremgår det af figur 3-8, hvordan de udvalgt parametersæt modellerer rejsetiderne relativt meget forkert. Alle parametersæt overestimerer rejsetiden fra 07:15 til 07:40. Specielt parametersættet fra vejreglen giver en alt for stor rejsetid i forhold til den oprindelige rejsetid fra DataFromSky, hvilket også fremgår af den samlede score, som er dobbelt så stor for vejreglens parametersæt i forhold til den genetiske algoritme. Fra 07:40 og fremefter underestimerer simuleringerne rejsetiden fra DataFromSky, hvor forskellen mellem DataFromSky og henholdsvis Brute force, GA og vejreglen er 54, 55 og 53 sekunder. Generelt viser resultaterne, at VIS-SIM har relativt svært ved at modellere spidsbelastningerne og især hvis der opbygges én model, som skal modellere



Figur 3-8 Rejsetid pr. minut fra motorvejen til Skærbækvej nord fra 07:15 til 08:15.

hele timen med samme værdier for de fem brugte kalibreringsparametre.

Hvor rejsetiderne for parametersættet, dannet på baggrund af vejreglens anbefalinger, er relativt svingende, så ligger både GA og Brute force relativt stabilt henover hele timen, selvom rejsetiden fra DataFromSky viser noget andet. Af

den årsag, er optimeringsalgoritmerne kørt på modellen fra tidspunktet 07:30 til 8:00, for at undersøge, om de vil komme med et andet parametersæt, som modellerer spidstimen bedst.



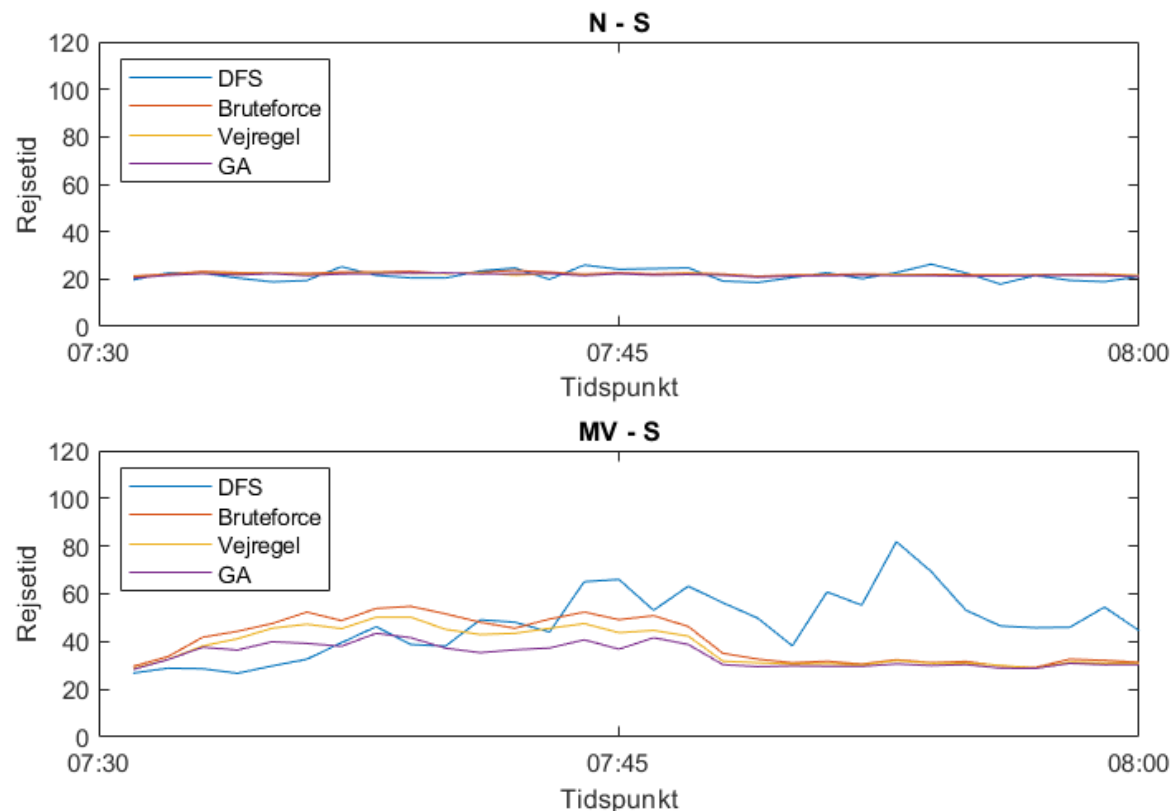
### 3.4 30 MINUTTERSMODEL

For at undersøge om algoritmerne finder andre, og eventuelt bedre, parametersæt ved at indsnævre søgeområdet til den kritiske halve time fra 07:30 til 8:00 opsættes Brute force og GA til at køre samme model igennem igen, men denne gang blot for de kritiske 30 minutter. De bedste parametersæt fra optimeringsalgoritmerne og vejreglen er vist i tabel 3-6:

| FG  | SF  | AX  | BX <sub>add</sub> | BX <sub>mult</sub> | Type |
|-----|-----|-----|-------------------|--------------------|------|
| 0,7 | 0,1 | 2,4 | 2,2               | 3,3                | BF   |
| 0,1 | 0,6 | 1,9 | 1,7               | 2,2                | GA   |
| 0,5 | 0,5 | 2,0 | 2,0               | 3,0                | VR   |

**Tabel 3-6** Bedste rejsetider for henholdsvis Brute force, GA og vejreglen. BF = Brute force. GA = Genetisk Algoritme. VR = Vejreglen for mikrosimulering.

Med de bedste parametersæt fra hver optimeringsalgoritme foretages igen VISSIM simuleringer, hvor rejsetider trækkes ud og præsenteres i forhold til hinanden og DataFromSky, som vist på figur 3-9. Der er kun præsenteret to ruter i dette afsnit, hvorimod resten er vist i bilag D.



**Figur 3-9** Rejsetid pr. minut for henholdsvis ruten fra Skærbækvej nord til Skærbækvej syd og motorvejen til Skærbækvej syd.

Det fremgår af figur 3-9, at VISSIM igen ingen problemer har med, at modellere rejsetider fra nord til syd, hvor der ikke er de store fremkommelighedsproblemer og en nogenlunde jævn rejsetid hen

over simuleringsperioden. Samtidig viser resultatet også, at VISSIM igen har svært ved at modellere virkeligheden, når systemet bliver belastet, og rejsetiden dermed øges i en kort periode.

Generelt kan det siges, at den kortere 30 minuttersmodel modellerer rejsetiden bedre, men der er stadig for store udsving i forhold til at modellere den korrekt, dog passer vejreglens værdier en del bedre for forkortede model.

## 4 DISKUSSION

*Formålet med indeværende kapitel er at sammenfatte og diskutere de i projektet anvendte metoder og resultater, som er dannet heraf. Efterfølgende vil det videre arbejde og muligheder blive belyst.*

### 4.1 FORUDSÆTNINGER/METODE

Projektets datagrundlag er relativt smalt, da det er baseret på én rundkørsel ved en motorvejsafkørsel. Det ville have været optimalt at inddrage flere forskellige rundkørsler, og eventuelt andre krydstyper, men da opbygningen af VISSIM netværk tager lang tid, er der i projektet kun benyttet én lokation. Ved andre rundkørsler eller krydstyper, kunne der være andre typer af ankomstfordelinger og andet adfærd, som vil kunne give andre resultater.

Til at evaluere rejsetid i DataFromSky indlægges snit ved samme længde, som data counters indlægges i VISSIM. Disse snit indlægges manuelt i DataFromSky, hvorefter længden fra stopstregen til

snittet måles manuelt i DataFromSky. Denne opmålingsproces bygger på DataFromSkys billedvridning, som er geokodet. På den måde antages billedet at være målfast, og der kan derfor måles distancer op direkte i DataFromSky. Selve opmålingen foregår manuelt med målepunkter, men dette kan kun gøres med en vis nøjagtighed. Derudover er det ikke sikkert, at vejene i VISSIM modellen ligger på præcis samme måde, som i DataFromSky, da vejene i VISSIM er opbygget manuelt efter et ortofoto, men det drejer sig om centimeterpræcision, hvilket ikke vurderes at have den store betydning.

Den opmålte rejsetid i DataFromSky er baseret på rejsetider fra én dag, hvilket gør datagrundlaget relativt lille. Det medvirker samtidig, at for visse ruter har der nogle minutter ikke været registreret nogen rejsetid. Hvis der havde været filmet over flere dage, ville det have været muligt, at skabe et mere nuanceret billede af den gennemsnitlige rejsetid for hvert minut. Dermed kunne der skabes flere modeller, for de forskellige dage, da dagsvariationen i trafikken

for en sådan model kan have relativ stor indflydelse på modellens resultater.

I VISSIM indlægges rutevalg og trafikmængder baseret på DataFromSkys målinger. Trafikmængderne indlægges med kvartersintervaller, hvorefter VISSIM danner en tilfældig ankomstfordeling. Det betyder, at VISSIM vil modellere forskellige ankomstfordelinger, hvor sandsynligheden for at finde en rute, hvor der indenfor et givent minutinterval ikke har været registreret nogle biler, og dermed ingen rejsetid, er ganske usandsynlig. Da rejsetiden fra VISSIM skal holdes op med DataFromSky interpoleres der, mellem kendte værdier for at have en fuld rejsetidsmatrice. Hvis der ikke er registreret rejsetider for de første eller sidste minutter, benyttes de nærliggende værdier til extrapolation. Det gør, at nogle af rejsetiderne er skabt på baggrund af interpolation eller extrapolation, hvilket kan give fiktive rejsetider.

DataFromSky er brugt til at danne en OD matrice, som herefter indlæses i VISSIM. I VISSIM indføres trafikmængder

geografisk relativt langt væk fra den modellerede rundkørsel (1-2 kilometer) i tilfælde af ophobning af trafik. Det vil sige, at det trafik, som er opmålt i DataFromSky, som udgangspunkt er en del tættere på rundkørslen i måleøjeblikket i forhold til hvordan det indlægges i VISSIM. I praksis betyder det, at hele spidsbelastningen vil komme en smule senere. I teorien burde afstanden mellem målesteder i DataFromSky og begyndelsen af VISSIM modellens vejlinks udregnes og trafikinputtet i VISSIM burde derfor korrigeres, således det optræder i modellen på det rigtige tidspunkt.

Som nævnt tidligere indlægges trafikmængder og svingbevægelser på kvartersinterval, hvor VISSIM selv laver en tilfældig ankomstfordeling. Med DataFromSky er det muligt, at få data fra hver enkelte bil, hvilket, hvis det blev implementeret i VISSIM, ville give et helt andet udgangspunkt i forhold til ankomstfordelinger. Ved at modellere hver enkelte bil, eller blot på minutsinterval, vil ankomstfordelingerne ikke blive nær så tilfældige, da variationen over et minut

er markant mindre end variationen over et kvarter.

Alle simuleringer i forbindelse med optimeringsmetoderne er foretaget i VISSIM med 5 runs ved forskellige *random seeds*. Vejreglen for mikrosimulering skriver, og den gængse kutyme hos COWI er, at der skal foretages op til 25 runs med forskellige *random seeds*, for at tage højde for variationen i ankomstfordelingerne. Hvis der skulle foretages så mange runs for hvert parametersæt, vil simuleringstiden blive 5 gange så lang, hvilket vil betyde, at antallet af forskellige parametersæt, som testes, skulle reduceres markant. Når det bedste parametersæt for hver optimeringsmetode er fundet, køres simuleringen igennem med 25 runs, da det er antallet af runs vejreglen anbefaler, og da det kun skal gøres få gange. I praksis betyder det, at optimeringsmetoderne kunne finde en parameteropsætning, som var den bedste efter 5 runs, men ikke den bedste efter 25 runs, da variationen i ankomstfordelingerne kunne have ændret resultatet. Det vurderes dog, at det ikke

har en særlig stor betydning, da den primære funktion er at sammenligne de forskellige parametersæt i forhold til den endelige score. Simuleringerne med de forskellige parametersæt er desuden kørt med samme *random seeds*, hvilket vil sige, at der er samme ankomstfordelinger for alle parametersæt ved de forskellige runs.

For metoden med *Brute force* er der blandt andet ændret på parametre vedrørende *Conflict Areas*. Her er det valgt, at værdierne er gældende for alle vigepligter i rundkørslen. Dette gøres udelukkende for at spare på simuleringstid, da det nedbringer antallet af variable fra 11 til 5, hvilket gør simuleringstiden markant mindre. Ved fem variabler, med en relativ god bærbar computer, hvis alle mulige parameteropsætninger skal simuleres, skal der simuleres i  $10^5 \times 5$  minutter, hvilket svarer til over 340 dage. Hvis alle mulige løsninger for 11 variabler skal simuleres, skal der simuleres i  $10^{11} \times 5$  minutter, hvilket tager det op mod 100.000 år. Hvis det havde været muligt at simulere med mere

computerkraft, som dermed kunne nedbringer simuleringstiden, ville det være optimalt at have ændret parameterindstillinger for hver Conflict Area, da trafikanterne højst sandsynligt, vil have forskellig adfærd ved de forskellige vigepligtssituationer i rundkørslen.

For at nedbringe simuleringstiden benyttes kun 400 runs, hvilket svarer til ca. 16 timer. I MATLAB generes der derfor en tabel med tilfældige parameterindstillinger indenfor givne intervaller for de enkelte parametre. Det betyder, at ikke alle mulige parameteropsætninger indgår i modellen. Det medvirker også, at det er tilfældig, hvor god Brute force metoden er, da den bedste parameteropsætning kan være indenfor de 400 runs. Grundet den tilfældige udvælgelse vil det være muligt at få vidt forskellige resultater, hvis Brute force køres to gange i træk.

For metoden med Genetiske algoritmer er benyttet et plug-in fra MATLAB, som selv håndterer crossover, mutation og generationer. Det er muligt i MATLAB at ændre på en række indstillinger i forhold

til algoritmen. De valg, der er truffet i forbindelse med algoritmen, har unægtelig indflydelse på resultaterne, men med den viden, jeg har haft, er de blot valgt efter, hvad jeg har kunnet læse mig frem har været "best practice". Med den rigtige viden og udgangspunkt samt mere erfaring i forhold til optimering generelt ville andre indstillinger muligvis have været valgt og yderligere optimeret på GA.

Alle optimeringsalgoritmerne er sat op til at evaluere på scoren, som er summen af alle forskellene kvadreret. Ved at benytte denne metode vil en model, hvor trafikken blot er skubbet med 5 minutter, blive straffet lige så meget, som en model, hvor trafikken ikke tilnærmelsesvis ligner virkeligheden.

## 4.2 RESULTATER

Resultaterne fra optimeringsalgoritmerne viser, at det er muligt at opsætte et system, som kan automatisere denne kalibreringsproces. Det er muligt med Brute force, som finder tilfældige værdier og prøver sig frem i blinde, hvilket

gør den meget tilfældig og på ingen måde er garant for et brugbart resultat. Samtidig er det muligt med en genetisk algoritme, som efter hver generation kommer tættere og tættere på det optimale parametersæt. En af forudsætningerne for optimeringsalgoritmerne er, at der maksimalt må simuleres i 16 timer. Genplottet, som er skabt i forbindelse med den genetiske algoritme, viser dog, at algoritmen ikke har konvergeret efter de 16 timer, da det ikke er de samme Elitechildren, som går igen over flere generationer.

Kalibreringsprocessen, som er opsat i dette projekt, finder det parametersæt, som bedst modellerer hele systemets rejsetid ved at ændre på et parametersæt af Driving Behavior og Conflict Areas som sættes konstant for den undersøgte time og som er gældende for alle benene i rundkørslen. Resultaterne kunne godt pege imod en kalibrering, som er afhængig af flere parametre end de fem parametre, som er valgt i dette projekt. Ved kun at sætte én værdi for samtlige Conflict Areas modelleres alle vigepligtssituationerne ens, hvilket ikke virker



Figur 4-1 Principskitse af Conflict Areas i VISSIM med henholdsvis individuelle parametersæt og ét samlet parametersæt.

plausibelt i forhold til den typiske adfærd, vi kender fra virkeligheden. Der burde opsættes et system, hvor de forskellige Conflict Areas kan varieres, således trafikken modelleres forskellige ved de forskellige vigepligter. Den eneste bagside ved dette er, at det kræver mere computerkraft eller længere simuleringstid, da antallet af ukendte variabler går fra 5 til 11, hvilket gør optimeringsalgoritmerne meget tungere at køre. Dette kunne løse problemerne i forhold til markant længere/kortere rejsetid for visse ben, da vigepligten dermed kan korrigeres så rejsetiden passer for alle ben. Netop disse betragtninger

om individuelle parametersæt for hver vigepligt er yderligere underbygget af (Ištoka Otković, Tollazzi and Šraml, 2013), til trods for at de har benyttet Priority Rules, som konkluderer følgende: *"It is recommended to consider parameters of minimum headway distance and minimum gap time for every entrance into an intersection separately."* I det videre arbejde vil det være relevant at inddrage netop disse overvejelser og undersøge, hvor stor betydning individuelle parametersæt for Conflict Areas har.

En anden løsning, eller kombination med forrige, kunne være at dele modellen op

efter tid i flere segmenter, således der dannes flere delmodeller, som hver især modellerer et givent tidsinterval. I stedet for at modellere hele timen fra 07:15 til 08:15 i en model, kunne der dannes fire modeller, som hver modellerede et kvarter. Der kan dog opstå problemer i forhold til at få ankomstfordelingerne til at passe i enderne af hver model, men det burde kunne løses ved at fylde modellen med et indledende kvarter, som ikke indgår i resultaterne. Derudover vil det være tidskrævende at opsætte flere modeller, som hver især skulle køres



igennem optimeringsalgoritmerne i minimum 16 timer for at finde det optimale parametersæt.

I stedet for at opbygge flere delmodeller kan der, ved hjælp af COM modulet i VISSIM, opbygges et system, hvor simuleringen pauses efter et fastsat tidspunkt. Herefter kan visse parametre såsom Driving Behavior ændres, og simuleringen startes igen. På den måde skal der ikke arbejdes med flere modeller, men det kræver en vis forståelse for kodning med COM modulet. Der er dog to problemer med denne metode, som kan være yderst afgørende for, hvor godt det virker i praksis. Det er kun visse parametre, som kan ændres imens simuleringen er aktiv. Det gælder for Driving Behavior, men ikke for Conflict Areas. Det betyder, at det kun er Driving Behavior, som kan ændres løbende, hvorimod Conflict Areas, vil forblive konstante. Det andet problem er, at hvis der skulle ændres på Driving Behavior fx fire gange i løbet af en simulering, vil det gøre antallet af ukendte variabler meget større. Hvis der både skal ændres på

Conflict Areas for hver vigepligtssituation og samtidig Driving Behavior hvert kvarter, vil det give 18 variable, som afhængig af interval, kan ændres mange gange. Uden en meget stærk computer eller en accept af længere simuleringstid giver det ikke mening, at opsætte så komplekse modeller, hvor der er så variabler, som skal testes igennem. Der kan dog stilles spørgsmålstejn ved, hvor vigtigt simuleringstiden er, så længe det er indenfor rimelighedens grænser. I størstedelen af tilfældene vil det være muligt at forlænge simuleringstiden til maksimalt et par dage eller op til en uge. Der skal i det videre arbejde undersøges betydningen af netop en forlænget simuleringstid for at kunne bestemme den optimale simuleringstid for at scoren kan nå at konvergere.

Til trods for en forøget kompleksitet i optimeringsalgoritmen, virker det umiddelbart ikke til, at VISSIM er i stand til at modellere trafikken korrekt, når systemet er tæt på kapacitetsgrænsen. Det er hvert fald ikke mine erfaringer fra dette projekt og tidligere projekter (Yde-

Nielsen, 2018). I forhold til litteratursøgningen er der delte meninger om, hvor god VISSIM er til at modellere virkeligheden. Uden at vide hvilket forudsætninger undersøgelseerne fra litteratursøgningen har haft i forhold til hvor belastede systemer de har brugt, så har de i (Mehar, Chandra and Velmurugan, 2014) ikke været i stand at få kalibreret modellen tilfredsstillende med den automatiske proces, hvor de i andre undersøgelser opnår et tilfredsstillende resultat.

#### 4.3 FREMTIDIG ARBEJDE

*I indeværende kapitel er metoden og resultaterne for projektet diskuteret, hvor diskussion næsten stiller flere spørgsmål end den besvarer. I det følgende vil der blive beskrevet hvordan der kan arbejdes videre med dette projekt og hvilke undersøgelser der skal til.*

Det første naturlige step, der vil være i forhold til at få optimeret disse algoritmer, vil være at undersøge, hvor stor betydning det har, at lave individuelle Conflict Areas for hver vigepligt. Jeg tror

på, at der ligger et potentiale i, at kunne ændre på adfærden for hver trafikant, men hvor stor betydningen er, vides ikke.

Dernæst kunne det være spændende, at opsætte et system, som kan ændre på indstillingerne for Driving Behavior i løbet af simuleringen. Således kan spidsbelastningen, som i dette projekt eksempelvis kun varer 30 minutter, have andre indstillingerne end det første og sidste kvarter af simuleringen. Ved at ændre på indstillingerne for Driving Behavior i løbet af simuleringen burde det være muligt at få rejsetiden til at passe bedre end hvad tilfældet har været i denne rapport.

I tillæg til de to ovenstående overvejelser kunne det være spændende at undersøge, hvor stor betydning simuleringstiden har i forhold til det endelige resultat for optimeringsalgoritmerne. I dette projekt har optimeringstiden været fastlåst til 16 timer, men det vides ikke, hvor stor betydning en forlængelse af simuleringstiden har.

## 5 KONKLUSION

*I følgende afsnit præsenteres konklusionerne på baggrund af undersøgelserne foretaget i projektet. Projektarbejdet har taget udgangspunkt i følgende problemstilling, som vil blive besvaret i følgende:*

*Hvordan opsættes et tillægsprogram til VISSIM, som selv, ved hjælp af maskinlæringsprincipper, kan kalibrere modeller ud fra forudbestemte distancemål? Hvilke kalibreringsparametre bør indgå, og hvor godt modellerer VISSIM modellerne virkeligheden?*

For at kunne besvare ovenstående problemformulering er der foretaget en litteratursøgning, som har til formål, at dække den nuværende viden indenfor automatisk kalibrering af VISSIM modeller. Hovedresultaterne af litteratursøgningen viser, at der har været mest succes med genetiske algoritmer, som baseret på Darwins evolutionsteori løser

dette optimeringsproblem. Ved genetiske algoritmer sammenlignes virkeligheden og model ved en såkaldt distance-mål (Measure of Performance), en værdi, som kan måles både i virkelighed og model. Det mest anvendte hertil er rejsetiden, hvilket derfor også er benyttet i dette projekt. I forhold til kalibreringsparametre antyder litteratursøgningen, at adfærdsparametre (Driving Behavior) har den største indflydelse på modellens resultat. I tillæg til litteratursøgnings resultater, har jeg valgt at arbejde med VISSIMs måde at modellere vigepligt, da der på nuværende tidspunkt ikke har været nogen undersøgelser af automatisk kalibrering, hvor Conflict Areas indgår. Det giver samlet set 5 kalibreringsparametre, tre fra Driving Behavior og to fra Conflict Areas.

Rejsetiderne, og alt andet virkelig data såsom trafikmængder, hastighedsfordelinger og rutevalg, er anskaffet ved hjælp af databehandlingsprogrammet DataFromSky, som giver mulighed for at indhente data fra trackede dronevideoer.

Ved hjælp af MATLAB og VISSIM COM-modul er der opsat et system, som kan kalibrere VISSIM modellen automatisk ved at ændre på de fornævnte kalibreringsparametre. Der er valgt relativ få kalibreringsparametre, da flere parametre ville forøge simuleringstiden markant. Der er benyttet to forskellige optimeringsalgoritmer, Brute force og genetiske algoritme (GA), som kan evaluere på forskellige parametersæt og hernæst finde det parametersæt, som modellerer rejsetiden tættest på virkeligheden. Optimeringsalgoritmerne danner for hver parametersæt en score, som evaluerer hvor god modellens rejsetid er i forhold til den virkelige rejsetid. Ud fra scoren fremgår det, at den genetiske algoritme, som samtidig er den mest komplicerede, finder den mindste forskel mellem modellen og virkeligheden og dermed det bedste parametersæt.

Ud fra det bedste parametersæt fra henholdsvis den genetiske algoritme og Brute force samt vejreglens anbefalede parametersæt er der trukket rejsetider ud fra VISSIM, som viser, at alle model-

ler har svært ved at modellere en spidstime, hvor rejsetiden henover timen ændrer sig markant. For de ruter, som er mest belastet er rejsetiden op mod fire gange mindre, end den er i virkeligheden. Det skyldes, at parametrene er kalibreret efter at passe bedst muligt på hele systemet og ikke blot for det enkelte ben. For at undersøge om det skyldes variationen i spidstimen, er simuleringsperioden indskrænket til de mest kritiske 30 minutter. Resultaterne viser dog, at VISSIM stadig er indstillet til at modellere trafikken for aggressivt, hvilket negligerer de høje rejsetider, som kun optræder i få minutter af gangen.

Hvis disse modeller skal gøres bedre, skal optimeringsalgoritmerne kunne håndtere flere kalibreringsparametre, således der eksempelvis kan skelnes mellem adfærden ved hver Conflict Area, så de ruter, hvor trafikanterne i dette tilfælde kører for aggressivt kan korrigeres. Samtidig viser resultaterne også, at den genetiske algoritme efter 16 timer ikke er konvergeret endnu, hvilket kunne tyde på, at simuleringstiden ikke er lang nok.

## 6 LITTERATURLISTE

Albrink, R. *et al.* (2010) *Anvendelse af mikrosimuleringsmodeller, Vejregel.*

Babinec, A. *et al.* (2015) 'Automatic Vehicle Trajectory Extraction for Traffic Analysis', *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XL(March), pp. 25–27. doi: 10.5194/isprsarchives-XL-3-W2-9-2015.

Bishop, C. M. (2006) *Pattern Recognition and Machine Learning, Journal of Chemical Information and Modeling.* doi: 10.1117/1.2819119.

Borgeskov, F. *et al.* (2015) *Trafikteknik Håndbog for Kapacitet og serviceniveau, Vejregel.*

COWI and Finansministeriet (2012) 'Trængselsindikatorer for hovedstadsregionen', (September), p. 20.

Dong, J. *et al.* (2015a) 'VISSIM Calibration for Urban Freeways', (InTrans Project 14-487), p. 3.

Dong, J. *et al.* (2015b) *VISSIM Calibration for Urban Freeways.* IOWA STATE UNIVERSITY.

Finansministeriet (2017) *Ny aftale baner*

*vejen for billigere og sikrere biler, Finansministeriet.* Available at: <https://www.fm.dk/nyheder/pressemeddelelser/2017/09/ny-aftale-baner-vejen-for-billigere-og-sikrere-biler> (Accessed: 6 March 2018).

Guido, G. *et al.* (2016) 'Evaluating the accuracy of vehicle tracking data obtained from Unmanned Aerial Vehicles', *International Journal of Transportation Science and Technology.* Tongji University and Tongji University Press, 5(3), pp. 136–151. doi: 10.1016/j.ijtst.2016.12.001.

Hellinga, B. R. (1998) 'Requirements For The Calibration Of Traffic Simulation Models', p. 12.

Higgs, B., Abbas, M. and Medina, A. (2011) 'Analysis of the Wiedemann Car Following Model over Different Speeds using Naturalistic Data', *3rd International Conference on ...*, pp. 1–22.

Ištoka Otković, I., Tollazzi, T. and Šraml, M. (2013a) 'Analysis of the influence of car-following input parameters on the modelled travelling time', *Tehnicki vjesnik*, 20(5), p. str. 919–925.

Ištoka Otković, I., Tollazzi, T. and Šraml, M. (2013b) 'Calibration of microsimulation traffic model using neural network approach', *Expert Systems with Applications*, 40(15), pp.

5965–5974. doi: 10.1016/j.eswa.2013.05.003.

Kim, S.-J. *et al.* (2005) 'Calibration of Microsimulation Models Using Nonparametric Statistical Techniques', *Transportation Research Record: Journal of Transportation Research Board*, 1935, pp. 111–119. doi: 10.3141/1935-13.

Lahrman, H. and Leleur, S. (1994) *Vejtrafik.* 1st edn. Polyteknisk Forlag.

Li, Z. *et al.* (2013) 'Calibrating VISSIM roundabout model using a critical gap and follow-up headway approach', *Transportation Research Board, 92nd Annual Meeting*, (May), p. 16.

Lownes, N. E. and Machemehl, R. B. (2006) 'VISSIM: A Multi-Parameter Sensitivity Analysis', (2001), pp. 1406–1413.

Lu, Z. *et al.* (2016) 'A video-based approach to calibrating car-following parameters in VISSIM for urban traffic', *International Journal of Transportation Science and Technology.* Tongji University, 5(1), pp. 1–9. doi: 10.1016/j.ijtst.2016.06.001.

Maryland Department of Transportation (2017) 'MDOT VISSIM Modeling Guidance', (August).

Mehar, A., Chandra, S. and Velmurugan, S. (2014) 'Highway Capacity Through

Vissim Calibrated for Mixed Traffic Conditions', *KSCE Journal of Civil Engineering*, 18(2), pp. 639–645. doi: 10.1007/s12205-014-0440-3.

Motamedidehkordi, N., Margreiter, M. and Benz, T. (2016) 'Shockwave Suppression by Vehicle-to-Vehicle Communication', *Transportation Research Procedia*. The Author(s), 15(December), pp. 471–482. doi: 10.1016/j.trpro.2016.06.040.

Norvig, P. and Russel, S. (2009) *Artificial intelligence: a modern approach*, Pearson. doi: 10.1017/S0269888900007724.

Park, B. (Brian) and Jongsun Won (2006) *Microscopic Simulation Model Calibration and Validation*.

Park, B. (Brian) and Qi, H. (Maggie) (2005) 'Development and Evaluation of a Procedure for the Calibration of Simulation Models', *Transportation Research Record: Journal of the Transportation Research Board*, 1934, pp. 208–217. doi: 10.3141/1934-22.

Park, B. (Brian) and Schneeberger, J. (2003) 'Microscopic Simulation Model Calibration and Validation: Case Study of VISSIM Simulation Model for a Coordinated Actuated Signal System', *Transportation Research Record: Journal of the Transportation Research Board*, 1856(03), pp. 185–192. doi:

10.3141/1856-20.

Poulsen, L. J. et al. (2012) *Rundkørsler i åbent land, Vejregel*.

PTV Group (2016a) 'PTV VISSIM 9 Introduction to the COM API', p. 40.

PTV Group (2016b) 'PTV VISSIM 9 User Manual', p. 1055.

Qin, G., Huang, X. and Chen, Y. (2017) 'Nested one-to-one symmetric classification method on a fuzzy SVM for moving vehicles', *Symmetry*, 9(4). doi: 10.3390/sym9040048.

RCE SYSTEMS (2017) 'DataFromSky User Guide', p. 26.

Rrecaj, A. A. and Bombol, K. M. (2015) 'Calibration and Validation of the VISSIM Parameters - State of the Art', *Tem Journal-Technology Education Management Informatics*, 4(3), pp. 255–269.

Siddharth, S. M. P. and Ramadurai, G. (2013) 'Calibration of VISSIM for Indian Heterogeneous Traffic Conditions', *Conference of Transportation Research Group of India*. Elsevier B.V., 104, pp. 380–389. doi: 10.1016/j.sbspro.2013.11.131.

Smith, C. (2014) 'US 231 Scottsville Road Scoping and Traffic Operations Study - VISSIM Development and

Calibration Report'.

Tettamanti, T. et al. (2015) 'Iterative Calibration of VISSIM Simulator Based on Genetic Algorithm', *Acta Technica Jaurinensis*, 8(2), p. 145. doi: 10.14513/actatechjaur.v8.n2.365.

Transport- Bygnings- og Boligministeriet (2017) 'Bekendtgørelse om flyvning med droner uden for bymæssigt område', *BEK nr 788*, 2017(1257), pp. 1–8.

Vejdirektoratet (2018) 'Nøgletal om vejtransport', p. 45.

Yang, C. and Mattingly, S. P. (2011) 'Microscopic Simulation Model (VISSIM) Calibration for Freeways with Managed Lanes', *Journal of the Eastern Asia Society for Transportation Studies*, 9, pp. 1351–1363.

Yde-Nielsen, A. (2018) *Betydningen af parameterindstillinger for rundkørsler*. Aalborg University.

Yu, M. and Fan, W. (David) (2017) 'Calibration of microscopic traffic simulation models using metaheuristic algorithms', *International Journal of Transportation Science and Technology*. Tongji University and Tongji University Press, 6(1), pp. 63–77. doi: 10.1016/j.ijtst.2017.05.001.



## A. LITTERATURSTUDIE

På baggrund af litteratursøgningen er der lavet en sammenfatning af det undersøgte. Materialet i denne sammenfat-

ning indeholder kun artikler, undersøgelser og rapporter, som har været med til at klarlægge og danne den teoretiske baggrund inden for emnet kalibrering af

VISSIM modeller og maskinlæring. Internetsider, programhåndbøger og lignende er ikke inkluderet i dette appendiks, da disse blot indgår som referencer i rapporten og litteraturlisten.

| Titel                                                                                                                                    | Resumé                                                                                                                                                                                                                                                                                                                             |
|------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Analysis of the influence of car-following input parameters on the modelled travelling time. (Ištoka Otković, Tollazzi and Šraml, 2013a) | Studiet omhandler kalibreringsparametre til VISSIM modeller. De har i studiet brugt en etsporet rundkørsel, hvor de analyserer følsomheden er en række kalibreringsparametre. Resultaterne viser, at kalibreringsdelen er yderst vigtig for at få et brugbart resultat. I dette studie er forfølgelsesmodellen Wiedemann 74 brugt. |
| Automatic vehicle trajectory extraction for traffic analysis from aerial video data. (Sensing <i>et al.</i> , 2015)                      | Artiklen omhandler primært hvordan DataFromSkys trafikantdetektering og tracking fungerer. Den giver et indblik i hvordan det fungerer i forhold til både trafikantdetektering med bl.a. positive og negative samples samt en dybere forklaring af tracking. Det danner yderligere baggrundsviden for processen i DataFromSky.     |
| A video-based approach to calibrating car-following parameters in VISSIM for urban traffic. (Lu <i>et al.</i> , 2016)                    | I artiklen benyttes en automatiseret proces med videooptagelser og tracking til at kalibrere forskellige parametre i VISSIM herunder hastigheder, acceleration, safety distance osv. I artiklen har de valgt kølængde som deres "kalibreringsmål". I denne artikel er forfølgelsesmodellen Wiedemann 74 brugt.                     |
| Calibrating VISSIM roundabout model using a critical gap and follow-up headway approach. (Li <i>et al.</i> , 2013)                       | Artiklen beskæftiger sig med kalibrering af VISSIM modeller, hvor rundkørsler indgår. Ved hjælp af radartechnologi, har de skaffet data vedrørende hastighed og acceleration og samtidig har de ved hjælp af observationer skaffet data omkring tidsgab. I artiklen er opbygget                                                    |

|                                                                                                                        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                                                                                                        | en kalibreringsguide, som er baseret på følsomhedsanalysen foretaget i projektet. I denne artikel er forfølgelsesmodellen Wiedemann 74 brugt.                                                                                                                                                                                                                                                                                                                                                    |
| Calibration and Validation of the VISSIM Parameters – State of the Art. (Rrecaj and Bombol, 2015)                      | Studiet er en sammenkædning af andre studier, hvor fremgangsmåden til kalibreringen er gennemgået. Der er undersøgt kalibreringsmuligheder, distancemål, kalibreringsparametre herunder adfærdsparametre og stedspecifikke parametre. Konklusionen er, at Genetiske Algoritmer er den mest brugte type kalibreringsmetode og yderst effektiv, især når der ikke er alt for mange parametre involveret.                                                                                           |
| Calibration of microscopic traffic simulation models using metaheuristic algorithms (Yu and (David) Fan, 2017)         | Artiklen beskriver flere forskellige kalibreringsmetoder til automatisk kalibrering af VISSIM modeller. Der er taget udgangspunkt i flere car-following parametre fra Wiedemann 99, hvor hver metode forsøger at finde den bedste kalibrering i forhold til virkelige resultater. I blandt de udvalgte kalibreringsmetoder er "genetic algorithm" (GA).                                                                                                                                          |
| Calibration of Microsimulation Models Using Nonparametric Statistical Techniques (KIM, 2005)                           | Studiet omhandler automatisk kalibrering med "genetic algorithm" (GA) for en hovedvej i Houston. Der er taget udgangspunkt i Wiedemann 74 parametre. Derudover benytter de statistiske tests, til at teste om de simulerede og observerede populationer er ens. Der er brugt rejsetid, som "kalibreringsmål".                                                                                                                                                                                    |
| Calibration of microsimulation traffic model using neural network approach (Ištoka Otković, Tollazzi and Šraml, 2013b) | Artiklen omhandler mulighederne for at kalibrere VISSIM modeller ved hjælp af neurale netværk. Der er taget udgangspunkt i en rundkørsel, hvorfor Wiedemann 74 parametre er udvalgt til kalibrering. Rejsetiden er benyttet som "kalibreringsmål".                                                                                                                                                                                                                                               |
| Calibration of VISSIM for Indian Heterogeneous Traffic Conditions (Siddharth and Ramadurai, 2013)                      | I artiklen foretages en følsomhedsanalyse af kalibreringsparametre, hvorefter der bagefter benyttes en genetic algorithm til at kalibrere modellen. Artiklen tager udgangspunkt i Wiedemann 74 parametre, som der er udført en følsomhedsanalyse på. Af følsomhedsanalysen kunne det konkluderes at hastighedsfordeling, sikkerhedstiderne og gap-tiderne var den mest følsomme. Artiklen konkluderer også at der med genetic algorithm kan laves mere præcise modeller end med standardværdier. |

|                                                                                                                                         |                                                                                                                                                                                                                                                                                                                                |
|-----------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Development and Evaluation of a Procedure for the Calibration of Simulations Models ((Brian) Park and [Maggie] Qi, 2005)                | Et notat om kalibrerings- og valideringsprocedure til VISSIM. Notatet omhandler metodikken indenfor kalibrering og har samtidig afsnit om følsomhedsanalyse, der udføres umiddelbart før der arbejdes med genetiske algoritmer. Ydermere er der taget udgangspunkt i en case, hvor forfølgelsesmodellen Wiedemann 74 er brugt. |
| Evaluating the accuracy of vehicle tracking data obtained from Unmanned Aerial Vehicles (Guido <i>et al.</i> , 2016)                    | Artiklen beskæftiger sig med tracking af dronedata og hvad der kan forventes i forhold til nøjagtighed. Samtidig har de beskrevet processen i forhold til automatisk fotogenkendelse, som bruges i forbindelse med tracking.                                                                                                   |
| Highway Capacity Through VISSIM Calibrated for Mixed Traffic Conditions (Mehar, Chandra and Velmurugan, 2014)                           | En undersøgelse af kapaciteten på en motorvej ud fra virkelig data og VISSIM resultater. Undersøgelsen at VISSIM overestimerer hastighed og kapacitet for motorvejen. Da det er motorvejskørsel er der brugt Wiedemann 99.                                                                                                     |
| Iterative Calibration of VISSIM simulator Based on Genetic Algorithm (Tettamanti <i>et al.</i> , 2015)                                  | I artiklen arbejdes der med Genetic Algorithm (GA), hvor de kalibrerer en makroskopisk model med floating car data.                                                                                                                                                                                                            |
| Microscopic Simulation Model (VISSIM) Calibration for Freeways with Managed Lanes (Yang and Mattingly, 2011)                            | Artiklen tager udgangspunkt i kalibreringen af motorvej med flere vejbaner og kalibreringen af denne. I denne artikel er der taget udgangspunkt i Wiedemann 99. Kalibreringsparametrene er nøje udvalgt og forklaret, hvor især adfærdsparametrene er benyttet. Til kalibreringen er der brugt genetiske algoritmer.           |
| Microscopic Simulation Model Calibration and Validation Handbook (Park and Jongsun Won, 2006)                                           | Håndbog udviklet af Virginia Department of Traffic, som forklarer hvordan de griber kalibrerings- og valideringsprincipper an. Fra følsomhedsanalyser og dermed udvælgelse af parametre til kalibrering med algoritmer herunder genetiske algoritmer.                                                                          |
| Microscopic Simulation Model Calibration and Validation: Case Study of VISSIM Simulation Model for a Coordinated Actuated Signal System | Artiklen tager udgangspunkt i en større hovedvej modelleret med bytrafik, som kalibreres manuelt med en struktureret tilgang, hvor hver ændring bliver beskrevet. Samtidig forklares kalibreringsparametrene løbende, hvor Wiedemann 74 og vognskiftparametre er anvendt.                                                      |
| Requirements for the Calibration of Traffic Simulation Models (Hellinga, 1998)                                                          | Denne artikel omhandler hvad der skal til for at kalibrere trafikmodeller. Fra data til modelopbygning frem til distancemål.                                                                                                                                                                                                   |

---

|                                                                             |                                                                                                                                                                                                                                                                                      |
|-----------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| VISSIM: A Multi-Parameter Sensitivity Analysis (Lownes and Machemehl, 2006) | Artiklen går i dybden med de ti adfærdsparametre i Wiedemann 99 modellen for at undersøge de forskellige parametres indflydelse og afhængighed i forhold til kapacitet. De finder i undersøgelsen ud af, at CC0 og CC1 er specielt følsomme for ændringer i forhold til kapaciteten. |
| VISSIM Calibration for Urban Freeways (Dong <i>et al.</i> , 2015a, 2015b)   | Studiet tager udgangspunkt i en hovedvej i USA, hvor der undersøges flere parametre fra Wiedemann 74 og arbejdes med kalibrering af disse parametre. Der er i dette studie ikke brugt nogen form for automatisk kalibrering.                                                         |

## B. REJSETIDER FRA DATAFROMSKY

I følgende bilag er alle rejsetider fra DataFromSky illustreret i en tabel. De røde værdier er interpolerede værdier.

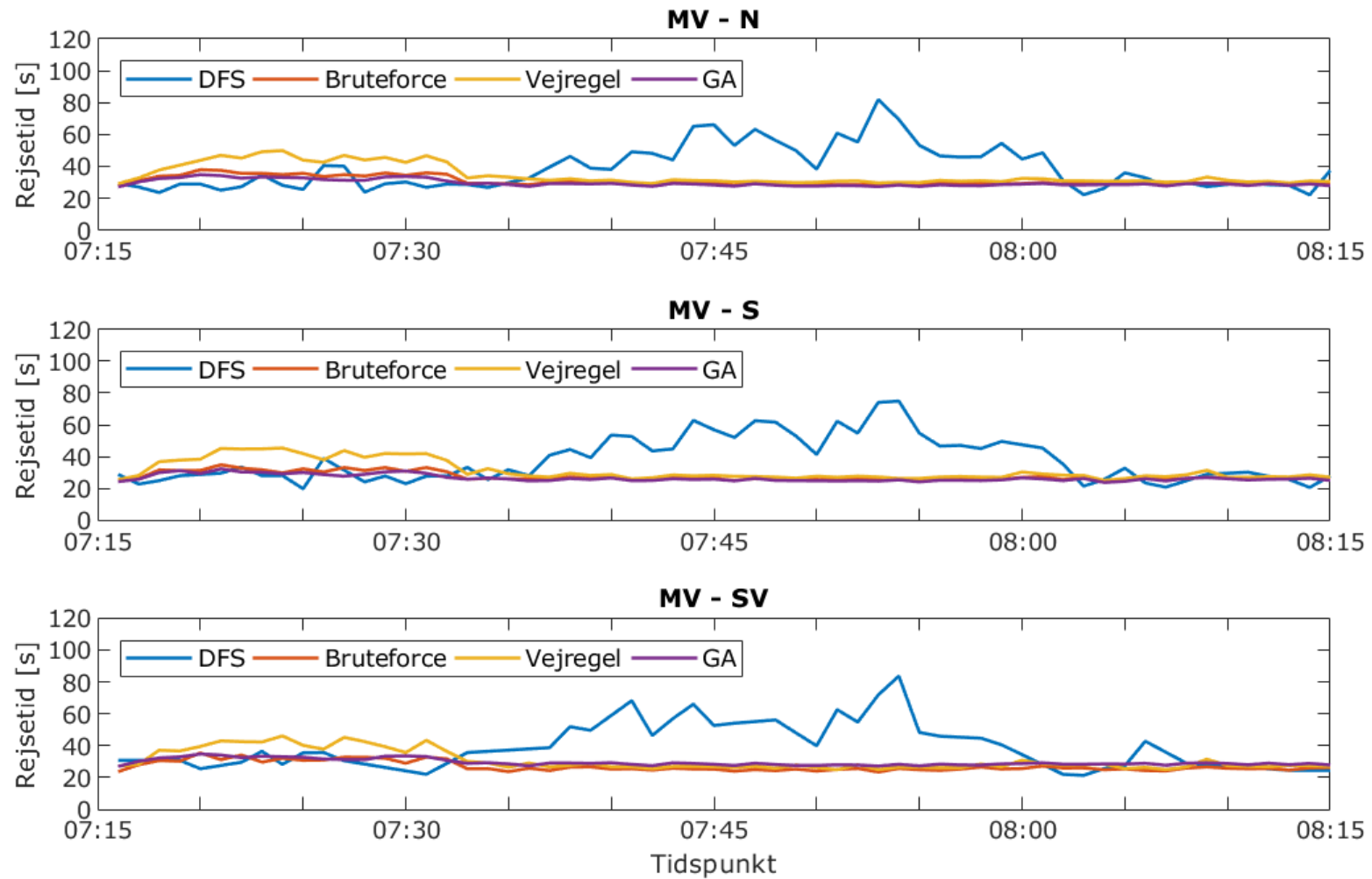
| MV_N | MV_S | MV_SV | N_MV | N_S  | N_SV | S_MV  | S_N   | S_SV | SV_MV | SV_N  | SV_S  | Tidspunkt |
|------|------|-------|------|------|------|-------|-------|------|-------|-------|-------|-----------|
| 29,3 | 28,9 | 30,8  | 16,9 | 21,3 | 17,7 | 20,2  | 25,0  | 35,9 | 24,9  | 35,5  | 48,7  | 07:15     |
| 27,0 | 22,8 | 30,8  | 16,9 | 21,7 | 18,7 | 25,6  | 21,2  | 35,9 | 33,8  | 35,8  | 48,7  | 07:16     |
| 23,5 | 25,0 | 30,8  | 17,3 | 18,6 | 16,8 | 31,1  | 28,0  | 35,9 | 24,8  | 28,2  | 48,7  | 07:17     |
| 28,9 | 28,1 | 30,8  | 17,7 | 24,0 | 18,7 | 26,3  | 29,2  | 35,9 | 39,5  | 35,8  | 54,1  | 07:18     |
| 28,9 | 28,8 | 25,6  | 18,1 | 22,9 | 21,9 | 27,2  | 26,3  | 35,9 | 33,7  | 62,0  | 59,4  | 07:19     |
| 25,0 | 29,6 | 27,6  | 18,5 | 19,1 | 18,7 | 28,0  | 28,9  | 35,9 | 32,0  | 25,2  | 49,1  | 07:20     |
| 27,2 | 33,5 | 29,6  | 20,8 | 15,8 | 19,1 | 15,3  | 22,9  | 35,9 | 30,3  | 22,2  | 38,7  | 07:21     |
| 34,6 | 28,1 | 36,5  | 23,2 | 19,5 | 19,4 | 29,0  | 34,3  | 35,9 | 28,5  | 36,3  | 28,3  | 07:22     |
| 28,1 | 28,2 | 28,3  | 25,5 | 19,0 | 17,2 | 14,2  | 38,1  | 35,9 | 26,8  | 34,4  | 35,2  | 07:23     |
| 25,5 | 19,8 | 35,6  | 27,9 | 25,9 | 19,8 | 27,4  | 20,8  | 35,9 | 23,2  | 23,3  | 42,1  | 07:24     |
| 40,5 | 38,9 | 35,7  | 30,2 | 23,1 | 22,4 | 40,6  | 28,6  | 35,9 | 38,2  | 49,5  | 49,0  | 07:25     |
| 40,1 | 31,5 | 30,6  | 26,7 | 22,7 | 19,4 | 53,8  | 51,0  | 35,9 | 53,2  | 59,2  | 55,9  | 07:26     |
| 23,8 | 24,2 | 28,5  | 23,1 | 20,5 | 18,3 | 18,5  | 36,1  | 35,9 | 46,7  | 54,6  | 62,8  | 07:27     |
| 29,1 | 27,9 | 26,3  | 19,5 | 19,3 | 18,4 | 32,8  | 30,7  | 35,9 | 16,3  | 33,9  | 58,8  | 07:28     |
| 30,2 | 23,1 | 24,2  | 15,9 | 26,3 | 18,4 | 47,2  | 46,1  | 64,2 | 27,3  | 32,7  | 54,8  | 07:29     |
| 26,8 | 27,8 | 22,1  | 17,5 | 19,7 | 17,8 | 42,8  | 40,9  | 56,8 | 21,4  | 28,4  | 50,8  | 07:30     |
| 28,8 | 27,9 | 28,9  | 19,1 | 22,6 | 17,2 | 33,2  | 28,8  | 49,4 | 44,7  | 48,1  | 46,8  | 07:31     |
| 28,6 | 33,2 | 35,7  | 18,8 | 22,5 | 16,6 | 23,9  | 23,5  | 52,3 | 38,4  | 34,8  | 49,9  | 07:32     |
| 26,7 | 25,6 | 36,5  | 18,5 | 20,5 | 21,9 | 14,6  | 17,0  | 50,3 | 24,0  | 24,8  | 44,1  | 07:33     |
| 29,8 | 31,8 | 37,2  | 18,2 | 18,8 | 18,1 | 15,1  | 21,2  | 48,3 | 30,7  | 24,1  | 38,3  | 07:34     |
| 32,6 | 28,1 | 38,0  | 17,8 | 19,4 | 17,9 | 35,4  | 25,4  | 46,4 | 37,4  | 39,0  | 32,5  | 07:35     |
| 39,5 | 40,9 | 38,8  | 17,5 | 25,2 | 17,8 | 29,9  | 21,2  | 44,4 | 44,1  | 25,7  | 26,8  | 07:36     |
| 46,2 | 44,5 | 51,9  | 17,0 | 21,6 | 19,9 | 24,3  | 20,9  | 42,4 | 50,8  | 54,3  | 52,9  | 07:37     |
| 38,8 | 39,4 | 49,7  | 16,4 | 20,5 | 15,5 | 26,9  | 37,2  | 41,3 | 51,3  | 51,0  | 57,1  | 07:38     |
| 38,1 | 53,6 | 59,0  | 18,6 | 20,5 | 24,0 | 29,5  | 41,0  | 40,1 | 61,9  | 64,3  | 61,2  | 07:39     |
| 49,1 | 52,7 | 68,2  | 20,8 | 23,4 | 18,2 | 32,1  | 45,4  | 38,9 | 72,4  | 79,9  | 65,3  | 07:40     |
| 48,1 | 43,6 | 46,4  | 19,8 | 24,7 | 23,8 | 65,1  | 51,9  | 73,2 | 101,8 | 104,3 | 69,4  | 07:41     |
| 44,0 | 44,8 | 57,0  | 18,7 | 19,9 | 18,8 | 98,0  | 102,5 | 74,4 | 73,5  | 68,3  | 73,6  | 07:42     |
| 65,1 | 62,7 | 66,0  | 18,7 | 25,9 | 21,1 | 125,2 | 117,6 | 75,6 | 41,7  | 44,8  | 74,4  | 07:43     |
| 66,1 | 56,9 | 52,6  | 18,6 | 24,2 | 23,9 | 100,1 | 107,9 | 76,9 | 54,1  | 74,3  | 75,2  | 07:44     |
| 53,1 | 52,0 | 54,1  | 18,6 | 24,4 | 20,2 | 75,1  | 82,2  | 78,1 | 66,5  | 80,2  | 88,1  | 07:45     |
| 63,2 | 62,5 | 55,1  | 19,2 | 24,7 | 19,0 | 95,6  | 99,7  | 79,4 | 67,8  | 60,4  | 101,0 | 07:46     |
| 56,2 | 61,7 | 56,2  | 19,7 | 19,2 | 22,1 | 73,7  | 86,8  | 80,6 | 72,6  | 75,0  | 92,0  | 07:47     |

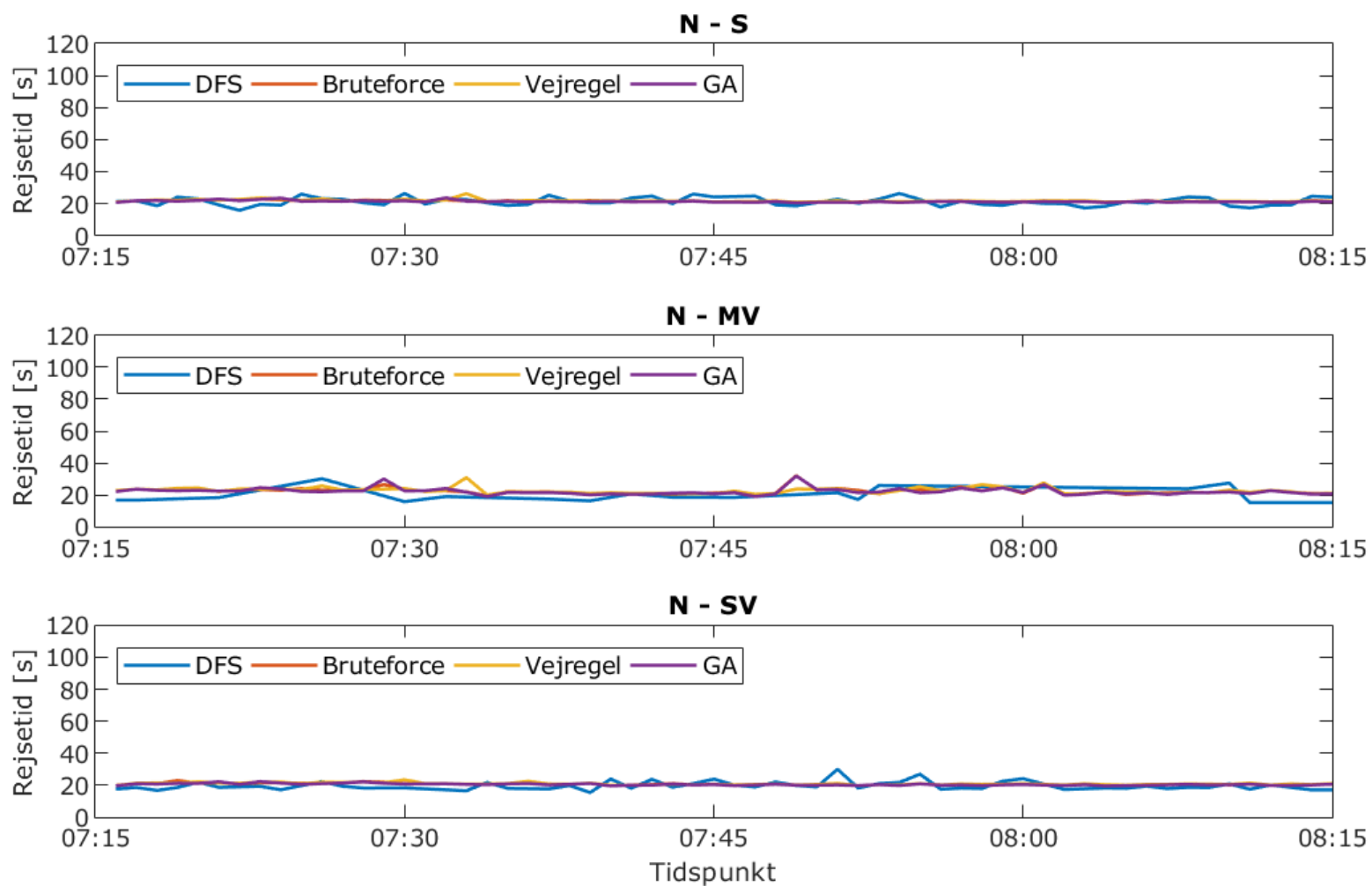
|      |      |      |      |      |      |      |       |      |      |      |      |       |
|------|------|------|------|------|------|------|-------|------|------|------|------|-------|
| 49,8 | 52,9 | 48,0 | 20,3 | 18,6 | 19,9 | 82,1 | 73,8  | 81,8 | 59,4 | 43,5 | 82,9 | 07:48 |
| 38,2 | 41,4 | 39,9 | 20,9 | 20,7 | 19,0 | 90,1 | 100,8 | 83,1 | 64,6 | 50,8 | 73,9 | 07:49 |
| 60,8 | 62,3 | 62,6 | 21,5 | 22,7 | 30,1 | 88,4 | 93,7  | 84,3 | 63,4 | 91,1 | 64,8 | 07:50 |
| 55,2 | 54,8 | 54,8 | 17,3 | 20,2 | 18,3 | 67,8 | 79,2  | 85,6 | 39,7 | 55,9 | 63,2 | 07:51 |
| 81,9 | 74,0 | 71,8 | 26,0 | 22,7 | 21,1 | 85,1 | 78,5  | 77,1 | 44,4 | 63,2 | 61,5 | 07:52 |
| 69,4 | 74,9 | 83,7 | 25,9 | 26,3 | 21,8 | 63,4 | 83,3  | 68,7 | 46,0 | 70,5 | 59,9 | 07:53 |
| 53,2 | 54,7 | 48,3 | 25,7 | 22,6 | 27,0 | 60,3 | 63,3  | 60,3 | 50,0 | 64,4 | 67,7 | 07:54 |
| 46,5 | 46,7 | 46,0 | 25,6 | 17,9 | 17,6 | 65,5 | 68,8  | 58,4 | 54,0 | 53,2 | 69,6 | 07:55 |
| 45,8 | 47,1 | 45,3 | 25,5 | 21,5 | 18,3 | 91,5 | 75,7  | 56,4 | 57,7 | 59,3 | 71,5 | 07:56 |
| 46,0 | 45,1 | 44,7 | 25,4 | 19,5 | 18,0 | 74,6 | 76,2  | 54,5 | 59,5 | 36,1 | 73,5 | 07:57 |
| 54,4 | 49,6 | 40,6 | 25,2 | 18,9 | 22,6 | 65,1 | 23,7  | 52,5 | 61,3 | 63,6 | 75,4 | 07:58 |
| 44,6 | 47,5 | 34,4 | 25,1 | 21,1 | 24,2 | 55,6 | 33,8  | 50,5 | 63,0 | 91,2 | 77,3 | 07:59 |
| 48,5 | 45,4 | 28,2 | 25,0 | 20,1 | 20,8 | 46,1 | 31,4  | 48,6 | 57,6 | 43,0 | 65,7 | 08:00 |
| 30,9 | 35,2 | 22,0 | 24,8 | 19,9 | 17,4 | 36,6 | 33,1  | 46,6 | 37,3 | 31,9 | 54,1 | 08:01 |
| 22,2 | 21,5 | 21,4 | 24,7 | 17,2 | 17,9 | 27,1 | 28,3  | 44,7 | 16,9 | 24,2 | 51,2 | 08:02 |
| 26,2 | 25,8 | 25,7 | 24,5 | 18,4 | 18,3 | 27,7 | 29,9  | 43,5 | 38,4 | 34,4 | 48,4 | 08:03 |
| 35,9 | 32,8 | 27,4 | 24,4 | 21,2 | 18,2 | 28,4 | 27,8  | 42,3 | 36,4 | 36,0 | 45,5 | 08:04 |
| 32,8 | 23,5 | 42,8 | 24,3 | 20,3 | 19,5 | 20,5 | 26,8  | 41,0 | 53,2 | 40,6 | 42,6 | 08:05 |
| 27,7 | 20,9 | 35,9 | 24,1 | 22,1 | 18,0 | 12,6 | 23,4  | 36,9 | 20,4 | 27,2 | 38,8 | 08:06 |
| 30,0 | 24,9 | 29,0 | 24,0 | 24,1 | 18,8 | 23,8 | 27,3  | 36,8 | 25,7 | 24,0 | 35,1 | 08:07 |
| 27,2 | 29,0 | 28,1 | 25,8 | 23,7 | 18,6 | 35,1 | 27,3  | 36,7 | 30,8 | 38,1 | 31,3 | 08:08 |
| 28,7 | 29,6 | 27,2 | 27,6 | 18,4 | 20,9 | 21,4 | 28,0  | 36,6 | 24,5 | 26,9 | 27,6 | 08:09 |
| 29,3 | 30,3 | 26,4 | 15,4 | 17,3 | 17,5 | 18,9 | 15,9  | 36,6 | 21,7 | 26,7 | 19,0 | 08:10 |
| 28,5 | 27,7 | 25,5 | 15,4 | 19,0 | 20,2 | 16,4 | 41,4  | 36,5 | 18,8 | 41,1 | 19,0 | 08:11 |
| 28,1 | 25,7 | 24,5 | 15,4 | 19,2 | 18,7 | 14,0 | 25,6  | 36,4 | 18,8 | 34,2 | 19,0 | 08:12 |
| 22,1 | 20,7 | 24,5 | 15,4 | 24,6 | 17,2 | 14,0 | 19,5  | 36,3 | 18,8 | 28,7 | 19,0 | 08:13 |
| 37,4 | 27,9 | 24,5 | 15,4 | 24,0 | 17,2 | 17,1 | 18,3  | 31,8 | 18,8 | 36,4 | 19,0 | 08:14 |

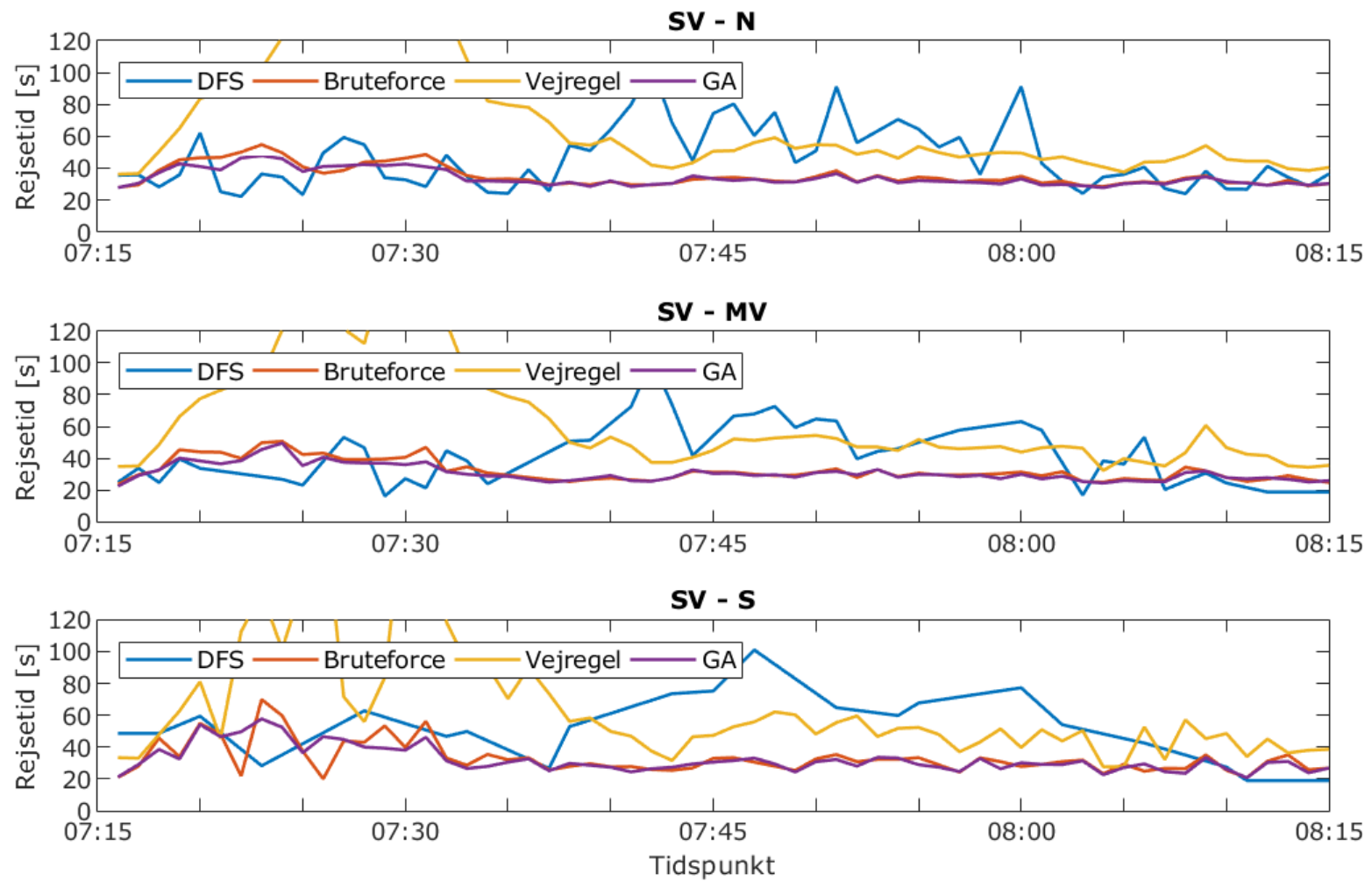


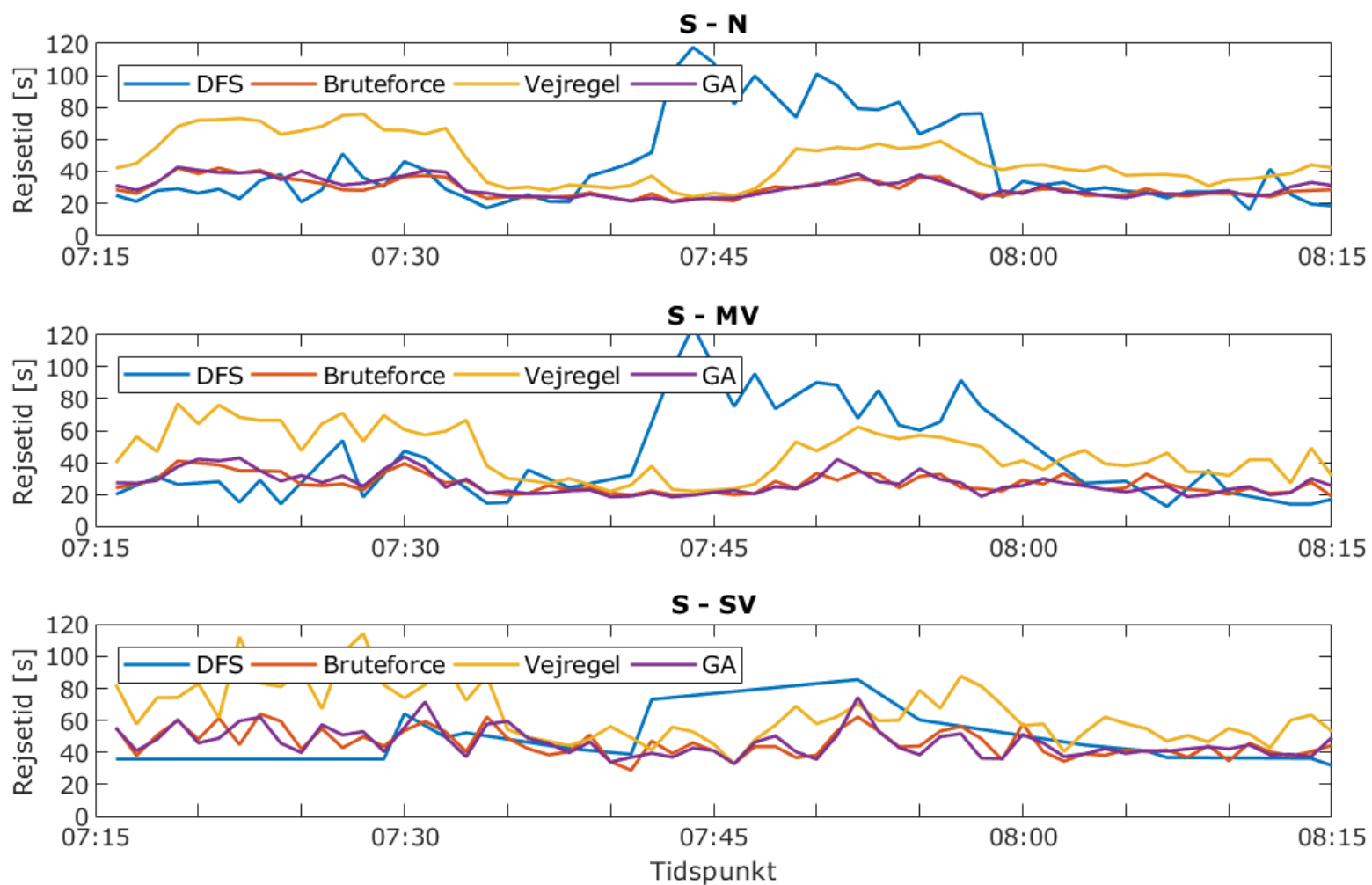
## C. RESULTATER FRA VISSIM ANALYSE

I følgende bilag er alle rejsetider fra VISSIM analysen illustreret ved grafer.









## D.RESULTATER FRA 30 MINUTTERSMODEL

I følgende bilag er alle rejsetider fra 30 minuttersmodellen illustreret ved grafer i forhold til rejsetider fra DataFromSky.

