
Simulering - Trafik - Temporary

Project Report
Group A319

Aalborg University
Det Teknisk-Naturvidenskabelige Fakultet
Strandvejen 12-14
DK-9000 Aalborg



AALBORG UNIVERSITY
STUDENT REPORT

**Det Teknisk-Naturvidenskabelige
Fakultet AAU**
Strandvejen 12-14
DK-9000 Aalborg
<http://cs.aau.dk>

Title:

Simulering - Trafik - Temporary

Theme:

Simulering

Project Period:

2. Semester 2016, P2

Project Group:

A319

Participant(s):

Benjamin Jhaf Madsen
Jacob Sloth Thomsen
Alexander Umnus
Kim Larsen
Lasse Fisker Olesen
Niclas Struntze Bach
Rasmus Thomsen

Supervisor(s):

Anders Mariegaard

Copies: 5

Page Numbers: 35

Date of Completion:

March 14, 2016

Abstract:

Pasta ipsum dolor sit amet rotini
pasta al ceppo lagane spaghetti
penne lisce tagliatelle conchiglie.
Stringozzi ricciutelle capellini lasag-
nette pennoni lasagnette trenette
croxetti capelli d'angelo mafalde
farfalle ziti strozzapreti rotini. Tagli-
atelle sacchetti pasta al ceppo
spaghetti foglie d'ulivo capunti
tortiglioni vermicelloni fettuccine.
Penne zita gnocchi manicotti sac-
chetti fiorentine corzetti pasta al
ceppo stringozzi vermicelli fusilli
lanterne sacchetti fettucelle. Fiori
tuffoli fiori tuffoli capelli d'angelo
sagnarelli chifferi tuffoli ricciolini
cavatappi.

Benjamin Jhaf Madsen
<bjma15@student.aau.dk>

Kim Larsen
<klars15@student.aau.dk>

Jacob Sloth Thomsen
<jsth15@student.aau.dk>

Lasse Fisker Olesen
<l lese15@student.aau.dk>

Alexander Umnus
<aumnus14@student.aau.dk>

Niclas Struntze Bach
<nbach14@student.aau.dk>

Rasmus Thomsen
<rkth15@student.aau.dk>

Forord

Læsevejledning

Terminologi

Contents

Preface	iii
1 Indledning	1
2 Metode	2
2.1 Brainstorm og problemtræ	2
2.2 Problemområde	2
2.3 Fremgangsmåde	2
2.4 Søgeprotokol	3
2.5 Kilder og kildekritik	3
I Problemanalyse	4
3 Teori	5
3.1 Trafik Model	5
3.1.1 Trafik Flow	5
3.1.2 Flow	6
3.2 Dijkstras Algoritme	7
3.3 INTRO TIL TEORI, SKAL SÆTTES I LØSNINGSAFSNIT . .	9
3.4 A* Algoritmen	9
3.4.1 Dijkstra og A* sammenligning	12
4 Problembeskrivelse	13
4.1 Problemets Relevans	13
4.1.1 Økonomi	13
4.1.2 Forurening	14
4.1.3 Livskvalitet	14
4.1.4 Nødsituationer	14
4.2 Eksisterende Modeller	14

5	Teknologianalyse	16
5.1	Indledning til teknologianalyse her	16
5.2	VisSim	16
5.2.1	VisSim - Bilen	16
5.2.2	VisSim - Netværket	17
5.2.3	Analyse af acceleration og deceleration	17
6	Altrans	19
6.1	Geografisk model	19
6.1.1	Model af kollektivt transportnet	19
6.1.2	Model af serviceniveau	20
6.1.3	Model for bilrejser	20
6.1.4	Model for attraktion til byfunktioner	20
6.2	Adfærds model	20
6.3	Emissions model	20
6.3.1	Varmstart	21
6.3.2	Koldstartstillæg	21
7	Interessantanalyse	22
7.1	Transport- og Bygningsministeriets (trm) - (Vejdirektoratet) . .	22
7.2	Kommunen	23
7.3	COWI	23
7.4	Visual Solutions	23
7.5	DTU	23
8	Problemformulering	25
II	Problemløsning	26
9	Løsningsforslag	27
10	Kravspecifikationer	28
10.0.1	Succeskriterier	28
10.0.2	•	28
11	Implementation	29
12	Diskussion	30
13	Konklusion	31
14	Perspektivering	32
	Bibliography	33
A	Appendix	35

Initierende Problem

Danske trafik modeller bliver ofte opbygget med et konkret formål i fokus og bliver svære at vedligeholde i fremtiden i de tilfælde hvor konteksten ændrer sig.

Arbejdsspørgsmål

- *Hvilke variabler bruges til at simulere trafik?*
- *Hvem har gavn af disse trafikmodellerings simulatorer?*
- *Hvordan fungerer de forskellige eksisterende software, trafik modellerings simulatorer?*

Dette afsnit har til formål at give et indblik i hvordan der er blevet arbejdet metodisk og videnskabeligt igennem projektet fra det initierende problem, og igennem rapporten til der opnås en endelige konklusionen. Metode kapitlet har til formål at gøre det muligt for læseren at få indsigt, hvordan vi igennem hele forløbet har arbejdet os frem til et færdigt produkt, og hvordan der overordnet er blevet arbejdet for at nå frem til dette produkt.

2.1 Brainstorm og problemtræ

Med udgangspunkt i hovede-emnet simuleringer var første proces, at undersøge mulige kategorier af under-emner, hvor man kunne se simulering som værende en del af en løsning til at afvikle et problem, samt et under-emne som et flertal synes kunne være interessant at arbejde med. Først blev gruppens medlemmer bedt at undersøge hvorvidt om det var muligt at finde noget brugbart materiale omkring de emner som de havde forslået og havde i tankerne, hvorefter de blev samlet i et problemtræ. Gruppens medlemmer fremlagde det som de havde fundet frem til, så det var muligt at vælge ved afstemning hvilket emne et flertal kunnes samles omkring til at indskrænke os.

2.2 Problemområde

Fra simulering blev det besluttet at indskrænke det yderligere til trafik simulering. Efterfølgende blev samme procedure benyttet under brainstormen til at afgrænse os yderligere indenfor trafik, således at det var muligt at vælge et meget afgrænset område, hvor det var muligt at finde materiale nok at arbejde med, til at lave en udførlige problemanalyse, hvorfra det var muligt at opstille en problemformulering der kan lede videre til en problemløsning.

2.3 Fremgangsmåde

Efter der blev gruppe

2.4 Søgeprotokol

2.5 Kilder og kildekritik

Når der blev fundet og benyttet kilder i vores projekt, blev det besluttet at alle kilder der måtte benyttes var kilder, som havde belæg i form af at det skulle være muligt at kunne kontakte vedkommende , eller have ophav fra statslige instanser eller anerkendte virksomheder og organisationer.

Part I

Problemanalyse

3.1 Trafik Model

Til dette projekt har vi valgt at arbejde med emnet trafik og simulering, mere specifikt simulering af trafik. Vi agter altså at løse et problem inde for dette område hvor i vores fokus ligger på at lave en simulering der kan hjælpe med til at konstruere og udspille forskellige scenarier der kan udspille sig i trafikerede områder og på den måde også simulerer alternativer. Derfor har gruppen valgt at udarbejde en model der beskriver gruppens fælles definition på trafik. Formålet med dette er at have en model at arbejde med og inddrage i programmet der fungerer som produktet i dette projekt.

OBS: Dette afsnit er ikke færdigt og modellen kan og vil med høj sandsynlighed ændre sig igennem projektet af forskellige årsager!

3.1.1 Trafik Flow

Trafik fenomener har i langt tid ikke været nemt at regne på. En publikation fra 1988 af Paul Ross fra Traffic Systems Division beskriver trafik som at have en vis lighed med væsker som ikke kan komprimeres mere end en vis densitet [16]. Igennem tiden har der været nogle forskellige teorier om hvorvidt man måler på trafik og mange har forsøgt på forskellige måder.

Den generelle konsensus for trafik variabler er følgende: Trafik Densiteten, K , farten, v , og volume, Q , er passende og brugbare til formålet beskrevet herover. [Note: Cite Dr. Henry Lieu]

$$Q = Kv \tag{1}$$

(3.1)

hvor følgende er gældende:

Q = trafik flow (Bil(er)/timen) forbi et punkt. K = vehicular densitet (bil(er)/km) v = (space-mean-speed) fart (km/t)

Densitet kan beskrives som antallet af fartøjer per længden af en enhed (i dette tilfælde km). De to vigtige former af densitet er kritisk densitet, K_c og jamdensitet, K_j . K_c er den maksimale densitet under free flow. K_j er den maksimale densitet under opkobning.

$$k = \frac{1}{s} \quad (2)$$

(3.2)

Hvor s er det inverse af densiteten, spacing, som er distancen fra midte til midte mellem fartøjer.

På en vej L vil densiteten K , på et bestemt tidspunkt t_1 , være lig det inverse af spacing mellem antal fartøjer.

$$K(L, t_1) = \frac{n}{L} = \frac{1}{\bar{s}(t_1)} \quad (3)$$

(3.3)

Space-mean-speed kan forklare som at være en udregning af fart hvori man tager et helt vejbane segment i betragtning. En serie af billeder eller video optager farten på individuelle fartøjer der kører på denne bane, ud fra dette er en gennemsnits fart udregnet. Denne type udregning anses for at være mere præcis end Time-mean-speed metoden som der ikke vil blive forklaret i dette afsnit. Udregningen for Space-mean-speed ser således ud:

$$v_t = n \left(\sum_{i=1}^n (1/v_i) \right)^{-1} \quad (4)$$

(3.4)

Hvor n er det antal af fartøjer der passerer vejbane segmentet.

3.1.2 Flow

Flow er det antal af fartøjer som passerer en form for reference punkt per enhed af tid, som fartøjer/timen. The inverse af flow er togfølge (h) hvilket er den tid der går imellem fartøjer der passerer det bestemte punkt og det forrige køretøj ($i + 1$). Ved overbelastning på veje forbliver h constant. Ved en trafik prop vil h gå mod uendeligt.

$$q = 1/h \quad (5)$$

(3.5)

Flow (q) der passere et bestemt punkt (x_1) i et interval (T) er lig det inverse af den gennemsnitlige afstand.

$$q(T, x_1) = \frac{m}{T} = \frac{1}{h(x_1)} \quad (6)$$

(3.6)

Lastbiler: Lastbiler og andre transport fartøjer er oftest skyld i at trafikken går langsommere eller i nogle tilfælde stopper helt op. Det er derfor at bl.a. transport af vindmølledele bliver igangsat sent om aften eller meget tidligt på morgenen således at de ikke skaber problemer for andre biler. Dette har altså en stor effekt på trafikken og kunne derfor være relevant at medtage i vores model, dette er dog ikke gruppens fokus da dette er et meget specifikt scenarie. NOTE: Kan ændres skulle vi have tid til at lave noget med denne type scenarie.

Tid på dagen / Rush Hour: Rush hour er det scenarie hvori der sker mest trafik i et land. Dette er typisk i de timer hvor de forskellige biler skal på arbejde, køre børn til skole eller andre institutioner eller lignende og igen når disse samme individer skal hjem igen. Dette kan indskrives i programmet som en form for variable der ændre mængden af trafik ved bestemte tidspunkter. NOTE: Mangler kilde på dansk rush hour.

3.2 Dijkstras Algoritme

Dijkstras algoritme er en algoritme til at finde den korteste vej fra et bestemt punkt til et andet punkt. Disse punkter kan blandt andet repræsentere den korteste vej mellem to forskellige byer. Dijkstras er en grådig algoritme, da den finder den korteste længde først og fortsætter således.[15]

Dijkstras algoritme finder den korteste rute mellem 2 forskellige punkter i en simpel ikke-orienteret vægtet graf. Man kan se på figur ?? at der angives forskellige punkter A, B, C, D, E, Z. Hvis man skal fra punkt A til Z på figuren, så starter man ved A og derfor initialiseres A til at være 0, som man kan se på tabel 3.1. Algoritmen virker således, at alle punkter er uendeligt udover det punkt man befinder sig på som vist på tabel 3.1. Algoritmen tager punkt fra punkt, så den starter med at se de grene som A har. Disse er $|AB| = 4$ og $|AC| = 2$. Her fra kan algoritmen ikke se videre end B og C. Den ser altid på det mindste tal, og derfor tager den længden fra A til B som er 4 og længden fra A til C som er 2, begge disse tal er mindre end uendeligt. Herefter ser den efter hvilket af de nuværende tal som er mindst, hvilket er 2. Så derfor vælger den C som sit næste punkt. Algoritmen finder nu de næst tætteste punkt, ved at addere alle de tidligere ruter, som har den korteste rute fra A til det næste sæt af punkter. Her ser algoritmen ud fra C og hvilke grene C har. Dette er længden til B, D og E, dog er længden altid fra A, så derfor er længden fra A til E 12 da $2+10 = 12$. Længden til B er nu blevet 3, da algoritmen ser på den

korteste rute, så A til B er 3, da $2+1 = 3$. Således fortsætter algoritmen indtil den rammer Z.[15]

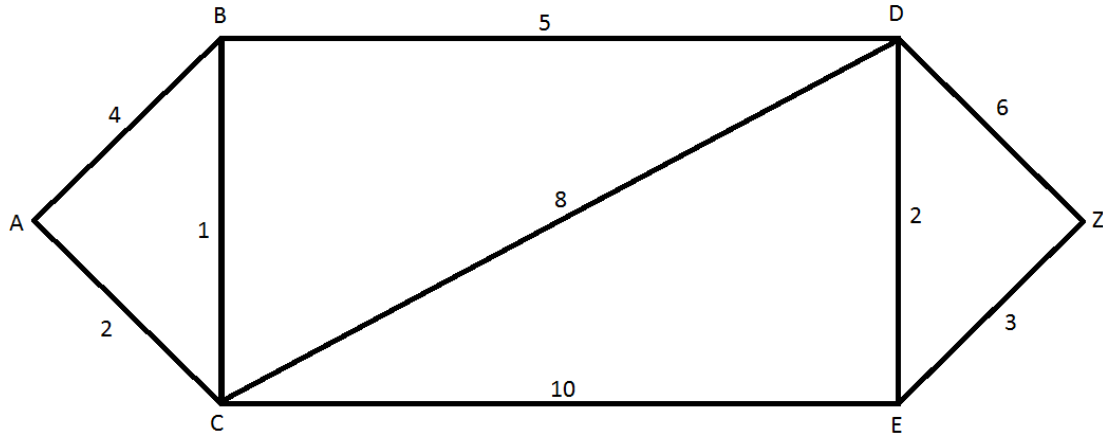


Figure 3.1: Graf til fremvisning af eksempel af Dijkstras algoritme i brug

	A	B	C	D	E	Z
	0	Inf	Inf	Inf	Inf	Inf
A	0	4	2	Inf	Inf	Inf
A, C	0	3	2	10	12	Inf
A, C, B	0	3	2	8	12	Inf
A, C, B, D	0	3	2	8	10	14
A, C, B, D, E	0	3	2	8	10	13
A, C, B, D, E, Z	0	3	2	8	10	13

Table 3.1: Dijkstra tabel

```

1  procedure Dijkstras(G: weighted connected simple graph, with ←
    all weights positive)
2  {G has vertices a = V0, V1, ..... Vn = z and lengths w(Vi, Vj) ←
    where w(Vi,Vj) = infinity if {Vi, Vj} is not an edge in G}
3  for (i = 1 to n)
4    L(Vi) = infinity
5  L(a) = 0
6  S = NULL
7  { the labels are now initialized so that the label of a is 0 and ←
    all other labels are infinity, and S is the empty set }
8  while (z does not belong to S)
9    u = a vertex not in S with L(u) minimal
10   S = S U {u}
11   for (all vertices v not in S)

```

```

12   if (L(u) + w(u, v) < L(v) then L(v) = L(u) + L(u, v))
13   {this adds a vertex to S with minimal label and updates the labels of vertices not in S}
14   return (L(z)) {L(z) = length of a shortest path from a to z}

```

Listing 3.1: Dijkstras angivet som eksempel i pseudo-kode

3.3 INTRO TIL TEORI, SKAL SÆTTES I LØSNINGS-AFSNIT

Til udformning af vores løsningsmodel bidrager dette afsnit en gennemgang af to algoritmer til ruteplanlægning og vejvisning. De er valgt på baggrund af deres udbredthed i industrien (KILDE). TEKNOLOGIANALYSEN anleder til at afgøre hvilke(n) algoritme(r) der er nødvendige til vores løsningsmodel. Der er blevet valgt to algoritmer til vejvisning som er A* og Dijkstras Algoritme. Der vil først blive gennemgået deres generelle funktion og dernæst en sammenligning af de to og en videre afgrænsning til hvilken der vil blive benyttet til løsningsmodellen.

3.4 A* Algoritmen

Primært når det kommer til belægning af en dynamisk rute, foregår det ved at en enhed fortsætter hen i mod et mål indtil den når en forhindring. Dette er et ekstremt simpelt bevægelsesmønster og indebærer in vis in-effektivitet. Rent retorisk kunne man stille spørgsmålet om det ikke ville være smartere at planlægge en rute før man overhovedet bevæger sig.

A* er en algoritme til at beregne den korteste rute baseret på en række heuristiske datasæt. A* får input igennem en brugerlavet graf der indeholder en række datasæt for at algoritmen kan fungere. Først har vi distancen fra punkt til punkt, eksempelvis punkt 'A' til punkt 'B' som vi kalder for f.eks. 'H' og dernæst har vi et datasæt 'G' der indeholder bekostningen for at flytte fra en kant til en anden, denne variabel er bestemt på forhånd. Et virkelighedseksempel kunne være at man vil over på den anden side af en sø, så har man så muligheden for at svømme direkte eller gå uden om og det koster f.eks. 2 gange så meget at bevæge sig direkte igennem søen. Dette er givet ved 'G', hvor som sagt 'H' er den ultimative korteste længde til det bestemte slutpunkt. 'H' fungerer desuden for hvilket som helst punkt i et system og angiver *altid* den korteste vej til slutpunktet uanset forhindringer. Det skal også nævnes at 'H' ikke er påvirket af bevægelsesbekostningen, til at starte med, som 'G' angiver, dette kommer først senere. Til sidst har vi 'F' der er en sammenlagt værdi af både 'H' og 'G'. Dette gælder kun for hver kasse der flyttes til, hvori 'H' er angivet ved kassen man flytter tils 'H' værdi. Det kan vises således i formlen 3.7:

$$F(n) = G(n) + H(n) \quad (3.7)$$

En måde man kan visualisere A^* på er f.eks. med et gitter-system som set i figur ???. Her kan vi se at vi har et start punkt (grøn) og et slutpunkts (blå). De kasser vi ikke kan bevæge os igennem er de røde kasser. Figuren angiver ingen heuristiske datasæt endnu.

1	2	3	4	5	6	7	8	9	10	11	12
13	14	15	16	17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32	33	34	35	36
37	38	39	40	41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70	71	72
73	74	75	76	77	78	79	80	81	82	83	84
85	86	87	88	89	90	91	92	93	94	95	96

Figure 3.2: A^* gitter-system

Ud fra figuren kan vi begynde os at forestille hvordan A^* fungerer. Når man bevæger sig fra kasse til kasse laver man 2 lister til at holde styr på hvor brikken har været. En liste til at holde styr på hvilke kasser man ikke har besøgt endnu og en liste der holder styr på hvilke man **har** besøgt. Når man flytter brikken skal man derfor angive hvilken kasse der nu skal på *besøgt* listen. Derfor som nævnt skal vi bruge information om hvor meget '**G**' koster. Brikken skal nu til at flytte sig for at komme til slutpunktet. Dette kunne f.eks. være 10 point for at flytte sig i hvilken som helst retning, men man kunne også sagtens angive at diagonal bevægelse ville koste 12 point. Dvs. at ruten ændrer sig til måske ikke at være så direkte som den ellers kunne have været.

Der findes flere metoder man kan anvende A^* på og en af dem vises her. Det vises her i den lille bid af Python-kode i Listing 3.2:

```

1 frontier = Queue()
2 frontier.put(start)
3 visited = {}
4 visited[start] = True
5

```



```

6 while not frontier.empty():
7     current = frontier.get()
8     for next in graph.neighbors(current):
9         if next not in visited:
10             frontier.put(next)
11             visited[next] = True

```

Listing 3.2: A stjerne og pseudo-kode af brug af lister

[14]

Som set i figur ?? har vi vores liste givet ved kassernes nummerering. Nummereringen kører fra venstre mod højre én række ad gangen. Vi angiver at det tager 10 point at gå lodret og vandret én kasse ad gangen og 12 point at gå diagonalt. I figur ?? kan vi nu se de heuristiske datasæt angivet fra startpunktet (grøn). Hver enkel kasse omkringliggende startpunktet har deres 'H' værdi angivet med lys-lilla tekst og bevægelsesomkostningen 'G' fra startpunktet til kassen angivet i blå tekst.

1	12	2	10	3	12	4	5	6	7	8	9	10	11	12
13	10	14	0	15	10	16	17	18	19	20	21	22	23	24
25	12	26	10	27	12	28	29	30	31	32	33	34	35	36
37	38	39	40	41	42	43	44	45	46	47	48	49	50	51
52	53	54	55	56	57	58	59	60	61	62	63	64	65	66
67	68	69	70	71	72	73	74	75	76	77	78	79	80	81
82	83	84	85	86	87	88	89	90	91	92	93	94	95	96

Figure 3.3: A* der viser bekostning af bevægelse fra startpunkt (grøn) til omkringliggende kasser ('G') angivet med blå farve samt 'H' angivet med lys-lilla

Nu udregnes 'F' værdien så f.eks. hvis vi går fra kasse 14 (startpunktet) til 15 skal vi lægge 10 ('G') og 8 ('H') sammen. Dette gør vi så for alle omkringliggende kasser for startpunktet. Dernæst går man til den laveste 'F' værdi og gør helt det samme som før, derudover flyttes den nye kasse man står på til *besøgt* listen. Noget man skal være opmærksom på her er at man stadig skal sammenligne bevægelsesomkostningen fra den tidligere kasse til de kasser der også er relevante for den nye kasse man har flyttet sig til. For dermed at afgøre om man kunne have påført en smartere bevægelse.

Denne fremgangsmåde er også bedre kendt som Breadth First Search, hvori

en frontlinje bliver kontinuerligt fremskyndet baseret på omkostninger og heuristiske datasæt[14]. A* er en heuristisk fremgangsmåde afledt af Dijkstras generelle funktionalitet.

3.4.1 Dijkstra og A* sammenligning

Problembeskrivelse

Trafiksystemet i Danmark undergår ofte udbygninger og ændringer, hvilket kan påvirke nærliggende vejnet. Det kan være svært at se hvordan disse ændringer vil påvirke trafikken, og derfor er der blevet opstillet forskellige modeller til at forudsige hvordan trafikken på vejnettet i fremtiden vil afvikle sig. Disse trafikmodeller bliver ofte opbygget med et konkret formål i fokus, og bliver derfor svære at vedligeholde i fremtiden i de tilfælde hvor konteksten ændrer sig. Konsekvensen af dette er at under en tredjedel af modellerne er blevet vedligeholdt, og at der ikke længere findes en model der dækker hele Danmark [17, s. 1-2].

4.1 Problemets Relevans

Formålet med trafikmodellerne er at forhindre trafikpropper og at sænke rejsetiderne. Uden modellerne er det besværligt at bestemme hvor der er problemer i vejnettet, og hvilken effekt nye veje vil have på trafikstrømmen. Trafikpropper har en effekt på landets økonomiske vækst, forurening, livskvalitet og tiden det tager for beredskaber og politi at nå frem.

4.1.1 Økonomi

Er man uheldig, kan man risikere at sidde fast i trafikken på vej til arbejdet. For at finde ud af hvilken effekt denne spildtid har på Danmarks økonomi, har Michael Knørr Skov og Karsten Sten Pedersen, der arbejder for konsulent firmaet COWI, analyseret 3 vejprojekter [12]. Vejprojekterne inkludere en tredje Limfjordsforbindelse, en ny motorvejsstrækning ved København, og en Forbindelse mellem Fyn og Als. Udfra COWI's beregninger vil disse tilføjelser spare danskere 25 tusinde timer dagligt, hvilket svarer omtrent til en værdi på 2500 millioner kroner årligt. Antager man at en fjerde del af denne tid bliver brugt på arbejde vil man opleve en BNP-vækst på 0,035%. Et velfungerende vejnet er dermed et vigtigt aspekt i forhold til at forbedre Danmarks økonomiske vækst.

4.1.2 Forurening

Billister er en af de største kilder af CO₂ forurening. Mængden af CO₂ der bliver udsluppet, afhænger af hastigheden billisterne kører. Ved en lav hastighed kan CO₂ udslippet per kilomet blive fordoblet, i forhold til at køre en stabil 50-130 km/t. I den anden ende, hvis man kører over de 130 km/t vil udslippet igen øges, da bilen er mindre effektiv i udnyttelsen af brændstoffet [11, s. 5-6].

4.1.3 Livskvalitet

En undersøgelse har vist at der er en sammenhæng mellem trafik densiteten, og stress niveauet på en individ der befærder sig i denne trafik. Udover at det kan være ubehageligt under kørslen, bliver stressen også ført med videre på arbejdet og til hjemmet. Stressen kan også føre til aggressiv kørsel og i værste tilfælde ender det med en ulykke [10, s. 2-3].

4.1.4 Nødsituationer

For den almindelige bilist, så kan trafikpropper være irriterende at skulle igennem, da det er tidskrævende. Men når det kommer til ambulancernes udrykning, og det kan have fatale konsekvenser for nogle patienter. Falck har oplyst at det koster ambulancerne 1-2 minutter i udrykningstid, når der er trafikprop. Konsekvenserne kan variere alt efter hvor alvorligt syg patienten er, og i værste tilfælde så er konsekvensen menneskeliv [13].

4.2 Eksisterende Modeller

De modeller der er vedligeholdet og stadig bliver brugt i dag, er meget forskellige i deres fokus. Der findes modeller som Senex, der analyserer godstrafikken mellem Danmark og Tyskland, der er en meget avanceret model til trafikafviklingen i hovedstadsområdet, og en masse mindre regionale og kommunale modeller [17, s. 2]. Forskellen på modellerne kan ses på detaljeringsgraden og hvor langt modellen kigger ud i fremtiden, hvor de mindre modeller har flere detaljer, men kun kigger få år ud i fremtiden, og vice versa for de større modeller. Trafikmodellerne er derfor delt op i 3 kategorier; strategiske, taktiske og operationelle modeller [17, s. 1].

Strategiske modeller er langsigtede modeller, men med færre detaljer. Manglen på detaljer er påkrævet, da det ellers vil blive for svært at anskaffe data'en, der skal bruges til at specificere alle forudsætningerne for modellens forudsigelser [17, s. 1]. Modeller af denne slags danner et billede over den internationale situation [17, s. 9]. Danmark benytter sig af en strategisk model, Trans-Tools, der blev udviklet i samarbejde med EU-kommissionen. Formålet med denne model er at forstå konsekvenserne af ændringer i det europæiske vejnetværk. Trans-tools hører også ind under taktiske modeller da den inkorporerer detaljer som for eksempel transportmiddelvalg [17, s. 10].

Taktiske modeller har i størstedelen af tilfældene et sigte mellem 3 og 20 år. I forhold til de strategiske modeller er detaljerings graden højere. Formålet

med disse modeller kan for eksempelvis være at finde ud af hvilke veje er belastede eller hvor lang tid en rejse vil tage [17, s1]. Taktiske modeller bliver brugt til at vise udviklingen i både internationale, nationale og regionale situationer [17, s. 9]. Modellerne der hører herunder er Senex, Storebæltsmodellen og Ørestadstrafikmodellen. Senex bruges til at vurdere tyske lastbilafgifter. Storebæltsmodellen bliver brugt til at vurdere takster og hvordan færgeudbuddet kan påvirke taksterne. Ørestadstrafikmodellen beskriver trafikken i Ørestad, og giver prognoser på hvordan en fremtidig stigning af antal biler vil påvirke vejnettet.

Operationelle modeller er kortsigtede modeller, og området man undersøger er meget afgrænset. Fordelen ved disse modeller er at den høje detaljerings grad kan give et mere præcist billede over situationen, dog kræver det at der skal bruges en masse data for at resultatet bliver realistisk [17, s. 1]. Herunder har vi kommunale og regionale modeller.

Udover disse er der mange andre modeller der blev opstillet, men ikke længere er vedligeholdte, herunder har vi eksempelvis Landstrafikmodellen, Hovedstadstrafikmodellen (HTM), national lastbilmodel, trafikafvilkningsmodeller og mange andre [17, s. 8]. Gennem problemanalysen vil en af de vedligeholdte modeller og en af de ikke vedligeholdte modeller blive undersøgt ved hjælp af en teknologianalysen, for at finde ud af hvilke elementer er vigtige. Informationen fra teknologianalysen kan derefter bruges til at lave en trafikmodel der er nem at vedligeholde.

5.1 Indledning til teknologianalyse her

5.2 VisSim

VisSim er et simulations program, som består af at man sætter blokke og diagrammer sammen, som former simuleringen. Det er en form for programmering, men man programmerer ved brug af blokke og diagrammer. VisSim benyttes for general modellering, simulation og designe simulations applikationer. Programmet bruges til at konstruere og simulere større dynamiske systemer. VisSim er programmeret i ANSI C, og under processen af et VisSim projekt kan projektet kompileres. VisSim er et diskret simuleringsprogram som modellerer adfærden for den enkle billist. VisSim benytter sig af psyko fysisk model, som benytter en regelbaseret algoritme ved bevægelser på tværs af banerne. Den psykologiske del bliver brugt til bilistens ønske om aggressivitet, hastighed, reaktionsevne og generelt menneskelige forhold til trafikken. Den fysiske del bruges til bilens adfærd, så som bilens hastighed, størrelse, position.

5.2.1 VisSim - Bilen

Den enkle bil spiller også en stor rolle hos VisSim, bilen er bestående af forskellige parametre, og det er ofte disse parametre der måles ved forsøg. Denne rapport tager udgangspunkt i acceleration og deceleration, ved vurdering af VisSim. Bilens parametre er beskrevet nedenfor.

- Ønsket acceleration.
- Deceleration.
- Acceleration
- Vægtfordeling.
- Hastighedsfordeling.
- Afstand mellem køretøjer.

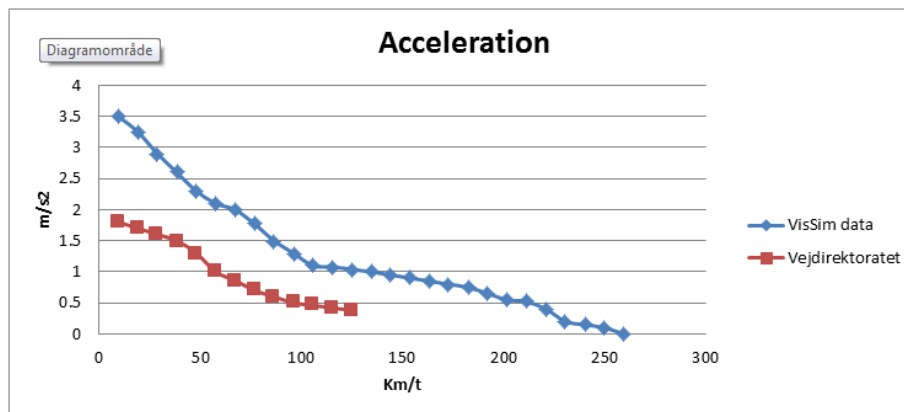


Figure 5.1: graf

- Størrelsen på køretøjet.

5.2.2 VisSim - Netværket

Netværket er bestående af de visuelle elementer, som har indflydelse på trafikafviklingen. Netværket parametre er beskrevet nedenfor.

- Rundkørsler.
- Vigepligt.
- Lyskryds(signalregulering).
- Hastighedszone.
- Vejbredde.
- Vejlængde.

5.2.3 Analyse af acceleration og deceleration

Ud fra en undersøgelse foretaget af Pihlkjær afgangsprøjet Aalborg Universitet - Vej og Trafikteknik, viser det sig at nogle af VisSims accelerations og decelerations værdier kan være upræcise. Undersøgelsen er foretaget ved analysering af VisSim på de danske vej-netværk. Undersøgelsen bruger data fra vejdirektoratet, dette data sammenlignes med VisSims accelerations data.

På figur ??, kan man se at accelerations fordelingen for VisSim er markant højere end dataen fra Vejdirektoratet. Dette viser sig, at være pga. VisSim er henvendt til de tyske-vejnetværk. I undersøgelsen beskriver de, at det skyldes de tyske biler er større og hurtigere. Dog bruger rapporten ældre data fra Vejdirektoratet, og derfor kan undersøgelsen også vise sig at være upræcis.

Der er yderlige undersøgelser foretaget af Pihlkjær afgangsprøjet Aalborg Universitet - Vej og Trafikteknik, hvor der sammenlignes med GPS accelerations og deceleration data, med VisSims data. Her er der blevet indsat GPS i 166 bilister som skal repræsentere acceleration og deceleration i Danmark. På figur

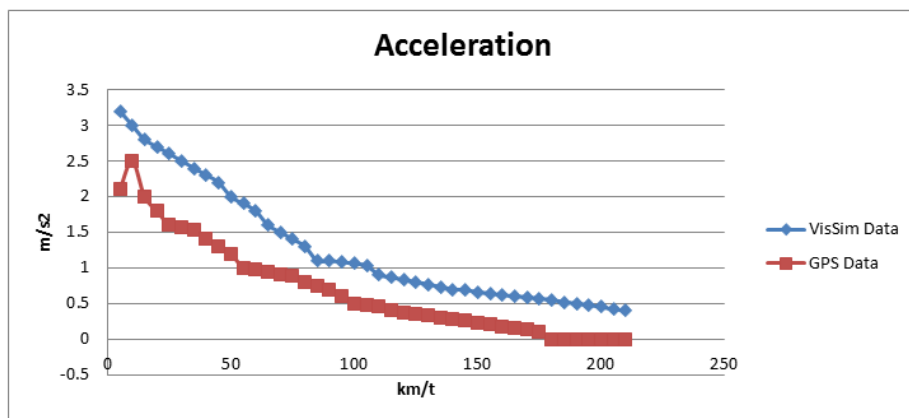


Figure 5.2: graf

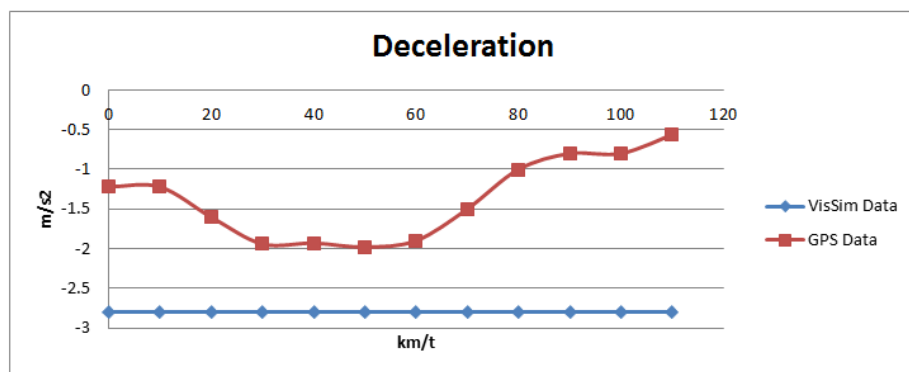


Figure 5.3: graf

?? kan man se, at det stadigvæk viser sig at VisSims acceleration er markant højere, end GPS daten. Dette kan give upræcis data, ved brug af simulering.

Samtidig er der undersøgt af Pihlkjær afgangprojekt Aalborg Universitet - Vej og Trafikteknik om deceleration for bilister er præcise i VisSim, dette er også gjort ved sammenligning af VisSim data, med GPS data. På figur ?? kan man se, at VisSims deceleration er markant højere end GPS dataen, man ser også at VisSim er lineært udspillet. Det vurderes at dette kan have betydning for udfaldet i simleringen, hvis VisSims data var nær GPS daten, så ville udfaldet blive mere præcist. Man ser også at VisSims data er konstant dette betyder, at programmet ikke variere decelerationen, i forhold til farten.

Alternativ Transportsystemer (Altrans) er en trafikmodel, hvis formål er at belyse hvordan en øget brug af den kollektive transport vil påvirke miljøet [8, s. 14]. Trafikmodellen er blevet udviklet af Danmarks Miljøundersøgelser (DMU) i 1994. Altrans består af 3 hovedmodeller: en geografisk model, en adfærds model og en emissions model [7, s. 14].

6.1 Geografisk model

Den geografiske model bruges til at beregne rejsetider, ventetider og skiftetider. For at udregne disse benyttes følgende undermodeller:

- Model af kollektiv transportnet
- Model af serviceniveau
- Model for bilrejser
- Model for attraktion til byfunktioner

Derudover benytter modellen sig af et geografisk informations system (GIS) til opbevaring af data og udregning af rejsetiderne [8, s. 18-19].

6.1.1 Model af kollektivt transportnet

For at lave realistiske simuleringer, bruger modellen for det kollektive trafik præcise data for ankomst- og afgangstider. Denne data kommer fra 11 trafiksel-skaber der bruger køreplanssystemet TR-System, DSB's data kommer i et andet format der bliver brugt til DBS's egen rejseplanlægger. Der er ikke brugt data fra Bornholm og kommunale ruter i Fyn, Region Midtjylland og Århus amterne, bortset fra ruterne i Århus, Odense og Randers. Modellen tager disse data og fortolker dem til et fælles format.

For at beregne rejsertiderne er kørestrækningerne, samt ankomst og afgangstiderne sat op i et tredimensionelt koordinatsystem, hvor tiden bliver indsat som den tredje akse Z. Stationernes placering bliver indsat som X og Y

koordinaterne, og man kan dermed finde ud af hvilke ruter der kan rejses med ved en given station. Efter at denne data er indsat kan modellen for beregning af rejsetider benytte [8, s. 20].

6.1.2 Model af serviceniveau

Serviceniveauet i Altrans bliver udregnet med variablerne tid, omkostninger og tilgængelighed. Modellen unlader at inkludere variabler som komfort, da DMU har lavet antagelsen, at komforten ikke ændre sig kraftigt over tid. Dette kan gøre prognoser der ser på den fjerne fremtid upræcise. Modellen spænder over både taktisk (meso) og operationel (mikro). På den taktiske plan kigger Altrans på buskilometer, afgangs frekvenser og tilgængelighed. På den operationelle plan kigges der på tiden man bruger i køretøjet, hvor lang tid man skal vente ved skift samt ventetiden i alt, og prisen på rejsen [7, s. 36-37].

6.1.3 Model for bilrejser

Formålet med modellen for bilrejser er at udregne tiden det tager at rejse fra by til by. Dette gøres ved brug af vejnettet i GIS, og hastigheden bilen kører kommer an på vejtypen. Ruten der bliver kørt starter og slutter fra centrumet af byerne der bliver rejst mellem. Modellen tager ikke højde for anden trafik på vejene, så tiderne der udregnes vil være præcise hvis der ikke er andre bliver på vejnettet. Hastighederne modellen bruger til de forskellige veje, kan ses på figur 6.1.

Motorveje	110
Motortrafikveje	90
Hovedveje	80
Øvrige veje på landet	70
Veje i byer	40

Table 6.1: Hastigheder på forskellige vejtyper

6.1.4 Model for attraktion til byfunktioner

6.2 Adfærds model

6.3 Emissions model

Man kan beregne emissioner af biltrafikken, dette gøres ved følgende: Der tages hensyn til og beregnes efter bilens tilstand, varm motor og koldstart. Det er en udregning der består ved at finde summen af en varstarts-emissionskoefficient gange trafikarbejdet og et koldstartstillæg for hver tur.

Disse faktorer er ikke selvstændige og bliver lavet per bilens årgang og dens størrelse/brændstoftype. Hastigheden af fartøjet determinere varmstarts-emissionskoefficienten. Der er sågar fortaget undersøgelser, såkaldte årskørselsundersøgelser af Vejdirektoratet (Winther & Ekmann, 1998) hvorfra det er konkluderet at årskørsel forudsat er uafhængigt af bilens størrelse selvom at dette

er set som urealistisk. Det estimeret trafikarbejde bliver udregnet ifølge en adfærdsmodel hvori man kigger på årskørsel pr bil i alle aldersgrupper. Der forøges eller reduceres med en faktor i selve fremskrivningsåret sådan at summen af antal biler i hver gruppe ganges deres gennemsnitlige årskørsel bliv lig det førnævnte trafikarbejde.

6.3.1 Varmstart

For at beregne varmstarts-emissionkoefficienten tager man udgangspunkt i COPERT II for den pågældende årgang. COPERT II er et windows program som gør det muligt at udregne emission fra vej trafik. Ydermere er programmet i stand til at lave lignende beregninger ud fra en forbrændingsmotor til et off-road fartøj. De beregnede emissioner inkluderer alle de store forurenende stoffer så som (CO , NO_x , VOC , PM) og mange flere. Programmet er ydermere i stand til at beregne brændstofforbrug [9, s 4]. For nye biler i alle fremtidsår anvendes der de emissionskoefficienter der er blevet udarbejdet som EU-normer pr den årgang. Dette er ikke gældende for ældre biler. Her ændre emissionskoefficienterne sig med tiden og dette er specielt pågældende for katalysatorbiler. Emissionskoefficienten bliver korrigeret fra årgang til årgang alt efter motorslid, da informationer om emissionskoefficienten udvikling afhænger af fartøjets samlede kørsel. Når den samlede årskørsel bliver udregnet er det også muligt at beregne den gennemsnitlige bils samlede kørsel for en givet alder. Motorslid korrigerede emissionskoefficienten ved varmstart kan således udregnes for hver bilårgang for et beregnings år, jf. Kveiborg(1999).

Til at beregne emissionen fra varmstarts er det forudsat at man har kendskab til fordelingen af trafik pr by-, land- og motorvejskørsel. ALTRANS har forudsat dette til at være som i basisåret.

6.3.2 Koldstartstillæg

I ALTRANS udregnes Koldstartstilægget

Interessantanalyse

Formålet med interessantanalysen er at identificere de interessanter som kan hjælpe med udvikling af produktet, det kunne f.eks. være i form af datadeling eller erfaringer. Vi vil i følgende afsnit kigge på nogle af de interessanter som vi synes kunne gavne vores projekt.

7.1 Transport- og Bygningsministeriets (trm) - (Vejdirektoratet)

Transport- og Bygningsministeriet(trm) er Danmarks øverste danske statslige myndighed på transportområdet og bygningsområdet. Trms hovedopgave er at sikre sig at de forskellige love bliver overholdt, ved opførelse af fx. en motorvej. Dog da trm er en sammensætning af mange underdelinger har vi valgt at fokusere på en af deres styrelser nemlig Vejdirektoratet.

Vejdirektoratet står nemlig bag statsvejnettet som hovedsageligt består af motorveje, hovedlandeveje, og mange af landets broer (i alt 3.801 km vej [18]. Det udgør med andre ord ca. 5% af det offentlige vejnet. Men selvom Vejdirektoratet kun står for 5% af vejnettet, så udgør disse 5% ca. halvdelen af Danmarks trafik.

Grunden til Vejdirektoratet kunne være interesseret i vores projekt ses på deres primære opgaver. Vejdirektoratet står nemlig for planlægningen af vejnettet så både privatbilismen og den kollektive trafik kan fungere sammen. Samt at de anlagte vej vil fungere i samspil med de allerede eksisterende veje. Vejdirektoratet vil være interesseret i vores simulering hvis den kunne gå hen og hjælpe med at forudsige placering af nye veje, så den virker i samspil med det allerede eksisterende vejnet. Altså at simulering kunne vise hvordan trafikken vil blive påvirket hvis der blev opført en vej.

En anden grund til Vejdirektoratet er valgt som en interessant er at de er en

styrelse som foretager mange undersøgelser/målinger omkring trafik. Vejdirektoratets undersøgelser vil derfor kunne bruges til udviklingen af vores produkt, og dermed formentlig gøre produktet mere realistisk.

7.2 Kommunen

Danmark har 98 kommuner som er delt over Danmark. Hver enkelt kommune står for de faktorer som påvirker kommunens areal, det kunne f.eks. være vedligeholdelse af veje eller opførelse af en. Kommunerne er derfor ofte midtpunktet i trafikprojekter vedrørende deres areal. De er derfor en vigtig interessant, da de er forbindelsen mellem de forskellige organisationer, f.eks. mellem Vejdirektoratet og Trafik- og Byggestyrelsen. Hvor Vejdirektoratet ofte står for planlægningen og Trafik- og Byggestyrelsen står for opførelsen. Kommunen er derfor ofte involveret i alle dele af projektet. Kommunen kunne derfor være en vigtigt interessant, da de har adgang til en masse undersøgelser/data som er foretaget i forbindelse med de forskellige trafikprojekter osv.

7.3 COWI

COWI er et internationalt rådgivningsfirma som har arbejdet med virksomheder overalt i verden, de var i 2015 involveret i ca. 13.000 projekter på verdensplanen [2]. COWI er en interessant da de har stor erfaring inden mange områder, heriblandt trafik. De har f.eks. rådgivet adskillige virksomheder i de sidste 30 år omkring, samt fremstillet trafikmodeller heriblandt for Vejdirektoratet, og flere kommuner [1]. COWI har altså en erfaring inden for fremstillingen af trafikmodeller, samt har de formentlig en masse datasæt som kan hjælpe os med at fremstille vores produkt.

7.4 Visual Solutions

Visual solution er firmaet der står bag VisSim, som blev grundlagt i 1989 [4], som er en matematisk simuleringsmodel. Da deres VisSim bliver benyttet af større internationale firmaer til at simulere og forbedre forskellige systemer. Over 100.000 forskere og ingeniører gør sig brug af VisSim når de skal arbejde med simulation, hvilket medfører til at firmaet er en interessant for vores projekt. Da de har meget erfaring indenfor simulering af trafikmodeller, da de sidder inde med en af de førende værktøj indenfor simulation [5]. De kan dermed være med til at bidrage til vores projekt.

7.5 DTU

DTU er et institut, som forsker inden for transportområdet. Deres mål er at skabe større vidensgrundlag for transport politiske beslutninger, hvorved de primært arbejder med optimering af trængsel i trafikken, miljøproblematikken, og trafikssikkerheden [6]. DTU har gennem årene foretaget en masse projekter, som vil kunne gavne og bidrage til vores projekt, f.eks. har de projekter vedrørende adfærdsmodeller, og trafikmodeller [3]. DTU vil altså kunne bidrage

med dataset og erfaringer omkring trafik, som vil kunne hjælpe med at øge realiteten i vores produkt.

8

Problemformulering

Part II

Problemløsning

9

Løsningsforslag

10

Kravspecifikationer

10.0.1 Succeskriterier

10.0.2 •

11

Implementation

12

Diskussion

13

Konklusion

14

Perspektivering

Bibliography

- [1] COWI. Trafikmodelreferencer. <http://www.cowi.dk/menu/project/0ekonomimanagementogplanlaegning/Trafikplanlaegningogmodellering/Pages/trafikmodelreferencer.aspx>, 2015.
- [2] COWI. Omcowi. <http://www.cowi.dk/topmenu/aboutcowi/Pages/omcowi.aspx>, 2016.
- [3] DTU. Dtu - projekter. <http://www.transport.dtu.dk/Forskning/Projekter?fr=1&mr=77&orgid=CTT&qt=DtuProjectQuery&pstatus=all#tabs>, 2016.
- [4] Altair Engineering. About visual solutions, incorporated. <http://www.vissim.com/company.html>, 2016.
- [5] Altair Engineering. Vissim. <http://www.vissim.com/>, 2016.
- [6] Carsten Broder Hansen. Dtu - forskning. <http://www.transport.dtu.dk/Forskning>, 2015.
- [7] Danmarks Miljøundersøgelser. Altrans - adfærdsmodel for persontrafik, faglig rapport fra dmu nr. 348. http://www.dmu.dk/1_viden/2_Publikationer/3_fagrapporter/rapporter/fr348.pdf, 2001.
- [8] Danmarks Miljøundersøgelser. Modelanalyser af mobilitet og miljø. slutrapport fra altrans og amor ii, faglig rapport fra dmu nr. 447. http://www.dmu.dk/1_viden/2_publikationer/3_fagrapporter/rapporter/FR447.pdf, 2003.
- [9] Leonidas Ntziachristos and Zissis Samaras. Computer programme to calculate emissions from road transport. <http://www.eea.europa.eu/publications/TEC05>, 1997.
- [10] Dwight A Hennessy og David L Wiesenthal. Traffic congestion, driver stress, and driver aggression. <https://www.researchgate>.

- net/profile/Dwight_Hennessy/publication/229863510_Traffic_congestion_driver_stress_and_driver_aggression/links/0deec53274dd4c9e88000000.pdf, 1999.
- [11] Matthew Barth og Kanok Boriboonsomsin. Traffic congestion and greenhouse gases. <https://escholarship.org/uc/item/3vz7t3db>, 2009.
 - [12] Michael Knørr Skov og Karsten Sten Pedersen. Trafikprop. flere veje vil skabe større vækst. http://www.cowi.dk/menu/tema/infrastruktur-2030/cowi-i-medierne/Documents/Veje%20skaber%20v%C3%A6kst_Politiken%20analyse%2024052014.pdf, 2014.
 - [13] Henrik Papsø. Trafikpropper kan koste menneskeliv. <http://www.tveast.dk/artikler/trafikpropper-kan-koste-menneskeliv>, 2007.
 - [14] Introduction to A*. <http://www.redblobgames.com/pathfinding/a-star/introduction.html>, 2016. Accessed: 06/03/2016.
 - [15] Kenneth H. Rosen. *Discrete Mathematics and its Applications*. McGraw Hill, 7. global edition, 20113.
 - [16] Paul Ross. Traffic dynamics. Technical Report 6, Traffic Systems Division, Federal Highway Administration, August 1988.
 - [17] Danmarks TransportForskning. Trafikmodeller arbejdsnotat til infrastrukturkommissionen notat 3. http://www.transport.dtu.dk/~media/Institutter/Transport/forskning/publikationer/publikationer%20dtf/2007/arbejdsnotat_om_trafikmodeller_160507.ashx?la=da, 2007.
 - [18] Vejdirektoratet. Længden af offentlige veje. http://www.vejdirektoratet.dk/DA/viden_og_data/statistik/vejeneital/1%C3%A6ngdeoffentligeveje/Sider/default.aspx, 2016.

A

Appendix