

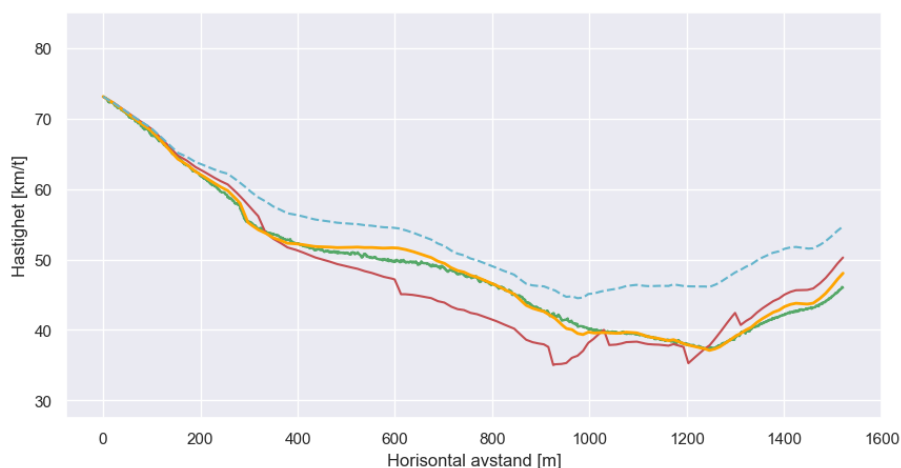
Jakob Kogstad og Simen Wrige Nygaard

Datadrevet utvikling av hastighetsmodell for tunge kjøretøy i stigning

Masteroppgave i bygg- og miljøteknikk

Veileder: Arvid Aakre

Juni 2021



Jakob Kogstad og Simen Wrige Nygaard

Datadrevet utvikling av hastighetsmodell for tunge kjøretøy i stigning

Masteroppgave i bygg- og miljøteknikk
Veileder: Arvid Aakre
Juni 2021

Norges teknisk-naturvitenskapelige universitet
Fakultet for ingeniørvitenskap
Institutt for bygg- og miljøteknikk



Kunnskap for en bedre verden

Sammendrag

Formålet med denne oppgaven har vært å videreutvikle en hastighetsmodell for tunge kjøretøy i stigninger. Eksisterende hastighetsmodell opererer med en generell antakelse om høy utnyttelsesgrad av konstant maksimal motoreffekt. I realiteten vil tilgjengelig motoreffekt påvirkes av flere faktorer, blant annet turtall, aktuelt gir og etterspurt dreiemoment.

Det er blitt gjennomført forsøkturer med lastebiler fra Scania, Mercedes-Benz og Volvo. Under forsøksturene ble posisjons- og motorloggerutstyr tilkoblet for logging av data. Innsamlet data fra forsøksturene har blitt brukt for å implementere variabel motoreffekt og girfaktor i en spesifikk hastighetsmodell for de ulike lastebilene.

Resultatene av en ny spesifikk hastighetsmodell viser et langt bedre tilpasset hastighetsprofil i forhold til de reelle forsøkturene, sammenlignet med eksisterende hastighetsmodell som benytter konstant motoreffekt. Eksisterende hastighetsmodell overestimerer tilgjengelige drivkrefter, som fører til høyere hastigheter enn hva som er tilfellet i virkeligheten.

Funnene i oppgaven indikerer at eksisterende hastighetsmodell kan gjøres mer kompleks for å beskrive virkeligheten med en større grad av nøyaktighet. Fremgangsmåten benyttet i oppgaven vil være av interesse for potensielle arbeider som fokuserer på ytterligere utvidelse av hastighetsmodeller for kjøretøy. Kartlagte muligheter rundt motordatalogging vil være nyttige i fremtidige arbeider som omfatter feltforsøk med tunge kjøretøy.

Abstract

The purpose of this work has been to improve an existing speed model for heavy vehicles in grades. A general assumption in the existing speed model is a high degree of utilization of constant maximum engine efficiency. Available engine efficiency will in reality among several factors be influenced by engine speed, gear and torque demand.

Experiments has been conducted with trucks from Scania, Mercedes-Benz and Volvo. During the experiments, equipment for logging position and engine data was applied. Accumulated data from the experiments has been used to implement varying engine effect and a gear factor in a specific speed model for the different trucks.

The results of a new specific speed model reveals significantly improved speed profiles in relation to the truck experiments, compared to the existing speed model that assumes constant engine efficiency. Existing speed model seems to overestimate available propulsion, leading to higher speeds than what occurs in reality.

The findings in this work implies that the existing speed model can be made more complex to depict reality with a higher degree of precision. The approach used in this work will be of interest for potential studies that focuses on further improvements of speed models for vehicles. Explored possibilities within engine data logging will be useful in upcoming studies that includes real life experiments with heavy vehicles.

Forord

Denne masteroppgaven er det avsluttende arbeidet på studiet bygg- og miljøteknikk ved NTNU i Trondheim, med transport som hovedprofil. Hovedveileder har vært Arvid Aakre, professor og leder for Trafikkteknisk senter ved NTNU.

Vi ønsker å takke Arvid for inspirasjon til oppgaven og innspill underveis. Avdelingsingeniør Jan Erik Molde har lagt inn mye tid i prosjektet, stilt opp som lastebilsjåfør under feltforsøk og generelt vært offensiv i planlegging av møter med transportbransjen i en samfunnsmessig utfordrende tid.

En stor takk rettes til Børstad Transport for gode faglige diskusjoner sentralt, samt sjåfører som har vært svært løsningsorienterte for å få til loggersekvenser på lastebiler i daglig drift. Scania, Mercedes-Benz og Volvo har hele veien spilt på lag med prosjektet, med tilgjengeliggjøring av tekniske data for enkeltkjøretøy.

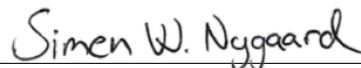
Statens vegvesen har vært behjelpelige med uthenting av vegdata, samt veiledning i bruk av NVDB-verktøy. Teknisk ansvarlig ved CSS Electronics har kontinuerlig bistått ved å svare på spørsmål som har vært rundt motorloggerutstyr.

Det rettes også en stor takk til kontor 2-276 for godt arbeidsmiljø og motivasjon gjennom semesteret. Det er flere som har bidratt på ulikt vis inn mot denne masteroppgaven, som ikke er nevnt direkte. All hjelp og støtte gjennom prosjektet settes stor pris på.

Trondheim, 09. juni 2021



Jakob Kogstad



Simen Wrigge Nygaard

Innhold

Sammendrag	i
Abstract	ii
Forord	iii
1 Introduksjon	1
1.1 Bakgrunn	1
1.2 Problemstilling	2
1.3 Prosjektoppbygging	2
1.4 Prosjektutførelse	3
1.5 Forkortelser og symbolforklaringer	4
2 Modellering av tunge kjøretøy	6
2.1 Motorkraft	6
2.2 Hastighet og turtall	6
2.3 Kjøretøydynamikk og drivkrefter	6
2.4 Akselerasjon	7
2.5 Kjøremotstand	7
2.5.1 Luftmotstand	7
2.5.2 Rullemotstand	8
2.5.3 Stigningsmotstand	10
3 Eksisterende arbeider	10
3.1 Børnes og Aakre (2011)	10
3.2 Statens vegvesens krav til forbikjøringsfelt i stigning	14
3.3 Tidligere forskning	16
3.3.1 Archilla og De Cieza (1996)	16
3.3.2 Bester (2000)	17
3.3.3 Rakha og Lucic (2002)	18
3.3.4 Lan og Menendez (2003)	19
3.3.5 Arellano, Echaveguren og Vargas-Tejeda (2014)	20
3.3.6 Opstad (2015)	21
3.3.7 Nesdal (2019)	21
3.3.8 Xin mfl. (2020)	22
3.4 Oppsummering	23
4 Datakommunikasjon i tunge kjøretøy	24
4.1 CAN-protokollen	24
4.2 SAE J1939-standarden	24
4.3 OBD - On-board diagnostics	26
4.4 FMS - Fleet Management System	27
5 Loggerinstrumenter	27
5.1 Posisjonslogging med VBOX fra Racelogic	28
5.2 Motordatalogging med CANedge1 fra CSS Electronics	28

6	Fremstilling og sammenstilling av loggerdata	31
6.1	Fremstilling av posisjonsdata fra VBOX	31
6.2	Fremstilling av motorloggerdata fra CANedge1	32
6.3	Fremstilling av høydeprofil	33
6.3.1	Høydeprofil med data fra NVDB	33
6.3.2	Høydeprofil med data fra VBOX	34
6.4	Sammenstilling av data til hastighetsmodell	34
7	Forsøk med lette kjøretøy	34
7.1	Utpøving av VBOX p� personkj�ret�y	34
7.2	Utpøving av CANedge1 med VBOX p� personkj�ret�y	35
7.3	Hensikt med pilotfors�k p� personkj�ret�y	36
8	Fors�k med tunge kj�ret�y	37
8.1	Fors�k med lastebil ved NTNU	37
8.1.1	Bakgrunn	37
8.1.2	Volvo FM 450, 2016-modell	37
8.1.3	Mercedes-Benz 2536L/45, 2012-modell	38
8.2	Fors�k med Scania fra B�rstad Transport	39
8.2.1	Bakgrunn	39
8.2.2	Scania R580, 2019-modell	40
8.2.3	Resultater	41
8.3	Fors�k med Mercedes-Benz i regi av NTNU	43
8.3.1	Bakgrunn	43
8.3.2	Mercedes-Benz Arocs, 2021-modell	43
8.3.3	Resultater	45
8.4	Fors�k med Volvo fra B�rstad Transport	48
8.4.1	Bakgrunn	48
8.4.2	Volvo FH 540, 2019-modell	48
8.4.3	Resultater	49
9	Sammenfatning av fors�k med tunge kj�ret�y	50
9.1	Datainnsamling	50
9.2	Hastighetssammenlikning	50
9.3	Teknologi for �konomisk kj�ring	51
10	Utvidelse av eksisterende hastighetsmodell	52
10.1	Form�let med utvidelse	52
10.2	Motbakkefremstilling	52
10.3	Motstandskrefter	56
10.4	Modellspesifikk hastighetsmodell	58
10.4.1	Sammenheng mellom hastighet og turtall	58
10.4.2	Girvalg	58
10.4.3	Variabel motoreffekt	59
10.4.4	Fremstilling	60
10.4.5	Scania	61
10.4.6	Mercedes-Benz	64
10.4.7	Volvo	67

10.4.8	Oppsummering av modellberegninger	68
10.5	Begrensninger i modellen	69
10.5.1	Horisontalgeometri	69
10.5.2	Kjøremotstand	69
10.5.3	Ytre omgivelser	69
10.5.4	Intelligente hastighetsprofil	70
10.6	Tilpassede girskift og motordata	70
10.7	Endring av utnyttelsesgrad i eksisterende modell	72
11	Oppsummering	74
11.1	Videre arbeid	75
11.1.1	Motordatalogging	75
11.1.2	Hastighetsmodell	75
Vedlegg A	Innlesning av CSV-filer	I
Vedlegg B	Stigningsberegning med NVDB-data	V
Vedlegg C	Stigningssammenligning VBOX-NVDB	VI
Vedlegg D	Listegenerering med iri- og mpd-data	VIII
Vedlegg E	Modellspesifikk hastighetsmodell	IX

Figurer

2.1	Relative lufthastigheter mot kjøretøy [Schramm, Hiller og Bardini 2014] . . .	8
3.1	Eksempel med inndata til hastighetsmodell [Børnes og Aakre 2011]	12
3.2	Eksempel med predikert hastighetsprofil fra hastighetsmodell [Børnes og Aakre 2011]	13
3.3	Eksempel med regnearkmodell for forbikjøringsfelt i stigning [Vegdirektoratet, Statens vegvesen 2019a]	15
3.4	Registrert mot modellert hastighet i stigning på 4,2 % [Archilla og De Cieza 1996]	17
3.5	Registrert mot modellert hastighet i stigning på 6,2 % [Archilla og De Cieza 1996]	17
3.6	Variabel drivkraft som en funksjon av hastighet [Rakha og Lucic 2002] . . .	19
3.7	Modellert akselerasjon ved implementering av variabel motoreffekt [Rakha og Lucic 2002]	19
3.8	Kalibrerte hastighetsprofiler for kjøretøy i motbakke < 1500 m, 100-160 kg/kW, 70-93 km/t [Arellano, Echaveguren og Vargas-Tejeda 2015]	20
3.9	Implementering av variabel motoreffekt i hastighetsmodell [Nesdal 2019] . . .	22
4.1	Deutsch HD10-9-1939-kontakt [Copperhill Technologies 2020]	25
4.2	J1962-kontakt med 12-voltssystem (egenprodusert)	26
4.3	J1962-kontakt med 24-voltssystem (egenprodusert)	26
4.4	FMS-utforming fra standard (egenprodusert)	27
5.1	VBOX Mini [Racelogic 2020a]	28
5.2	Loggerinstrumentet CANedge1 [CSS Electronics 2020b]	29
5.3	Adapterkabel av type 2, DB9 til Deutsch 9-pin [CSS Electronics 2020c] . . .	29

5.4	Adapterkabel DB9 til OBDII [CSS Electronics 2021c]	29
5.5	CANCrocodile med ekstern strømtilkobling fra CSS Electronics [CSS Electronics 2021a]	30
5.6	Datagang fra logging til fremstilling [CSS Electronics 2020d]	30
6.1	Eksempel på datafremstilling med VBOX Test Suite (Skjermbilde fra VBOX Test Suite)	31
6.2	Eksempel på datafremstilling med asammdf GUI (Skjermbilde fra asammdf GUI)	33
7.1	Sammenstilt data fra VBOX og Torque pro fra pilotforsøk med personkjøretøy	35
7.2	Kart med logget posisjonsdata fra første pilotforsøk med VBOX Mini (Skjermbilde fra Google Earth)	35
7.3	Kurvetilpasning av hastighetsprofiler fra loggerforsøk med personbil, CANedge1 og VBOX	36
8.1	Volvo FM 450, 2016-modell (Foto: Nygaard og Kogstad)	38
8.2	Bilde av Volvo FM 450, 2016-modell etter demontering av deksel på førerside (Foto: Nygaard og Kogstad)	38
8.3	Mercedes-Benz 2536L/45, 2012-modell (Foto: Nygaard og Kogstad)	39
8.4	Scania R580, 2019-modell med henger (Foto: Nygaard og Kogstad)	40
8.5	Uttevslingskurver for girskift, Scania R580, 2019-modell [Scania 2019]	42
8.6	Dreiemomentkurve, Scania R580, 2019-modell [Scania 2019]	42
8.7	Effektkurve, Scania R 580, 2019-modell [Scania 2019]	43
8.8	Mercedes-Benz Arocs, 2021-modell (Foto: Nygaard og Kogstad)	44
8.9	CANCrocodile montert i forbindelse med FMS-system (Foto: Nygaard og Kogstad)	45
8.10	Girutevslingskurve, Mercedes-Benz Arocs, 2021-modell [Mercedes-Benz 2021]	46
8.11	Effekt- og dreiemomentkurve, Mercedes-Benz Arocs, 2021-modell [Mercedes-Benz 2021]	47
8.12	Volvo FH 540, 2019-modell med henger (Foto: Nygaard og Kogstad)	48
8.13	Effekt- og dreiemomentkurver, D13K-motor fra Volvo [Volvo 2021]	49
9.1	Hastighetsprofiler fra loggerforsøk med Scania, VBOX og CANedge1	51
10.1	Høydeprofiler fremstilt fra NVDB- og VBOX-data	54
10.2	Felles nullpunkt for sammenligning av høydeprofiler fra NVDB- og VBOX-data	54
10.3	Sammenligning av stigning for NVDB- og VBOX-data	54
10.4	Kartutsnitt fra Google Earth over utvalg av Okstadbakken med start- og sluttkoordinat for loggerdata fra VBOX og NVDB (Skjermbilde fra Google Earth)	55
10.5	Eksempelbakke med hastighet og kjøremotstandsfordeling	57
10.6	Konstruerte moment- og effektkurver for lastebilene benyttet under forsøk	59
10.7	Spesifikk hastighetsmodell, Okstad, Scania R580	62
10.8	Spesifikk hastighetsmodell, Sveberg, Scania R580	62
10.9	Spesifikk hastighetsmodell, Vuddu, Scania R580	63
10.10	Spesifikk hastighetsmodell, Okstad, Mercedes-Benz Arocs	65
10.11	Spesifikk hastighetsmodell, Moholt, Mercedes-Benz Arocs	65
10.12	Spesifikk hastighetsmodell, Klett, Mercedes-Benz Arocs	66
10.13	Spesifikk hastighetsmodell, Okstad, Volvo FH 540	68
10.14	Hastighetsmodell med tilpasset giring, Okstad, Scania R580	71
10.15	Hastighetsmodell med tilpasset giring, Vuddu, Scania R580	71

10.16	Hastighetsmodell med justert utnyttelsesgrad, Okstad, Scania R580	73
10.17	Hastighetsmodell med justert utnyttelsesgrad, Vuddu, Scania R580	73

Tabeller

1.1	Kapitler i masteroppgaven som er dekket i forprosjektet (Nygaard og Kogstad 2020)	3
1.2	Forkortelser med beskrivelse	4
1.3	Symbolforklaringer med tilhørende enhet	5
2.1	Utvalg av kjøretøytyper med tilhørende dragkoeffisient, C_D [Rakha, Lucic mfl. 2001]	8
3.1	Statiske inputparametere for hastighetsmodell [Børnes og Aakre 2011]	11
3.2	Dynamiske inputparametere for hastighetsmodell [Børnes og Aakre 2011] . .	11
3.3	Dimensjonerende verdier som inngår i beregning av akselerasjon for vogntog [Vegdirektoratet, Statens vegvesen 2019b]	14
4.1	PGN-data [Copperhill technologies 2020]	25
4.2	SPN-data [Copperhill technologies 2020]	25
5.1	Tekniske spesifikasjoner tilhørende CSS Electronics' CANedge1 [CSS Electronics 2020a]	28
8.1	Nøkkelparametere tilhørende Scania R580, 2019-modell	40
8.2	Utvalg av relevante gjenkjente PGN-er etter loggerforsøk med Scania	41
8.3	Nøkkelparametere tilhørende Mercedes-Benz Arocs, 2021-modell	44
8.4	Nøkkelparametere tilhørende Volvo FH 540, 2019-modell	49
10.1	Utvalg av Okstadbakken med stigningsdata fra VBOX og NVDB	55
10.2	Lastebilmodellenes forholdstall mellom turtall og hastighet for toppgirene . .	58
10.3	Lastebilmodellenes operasjonelle turtallsområder	59
10.4	Motbakkeinformasjon ved beregning med modellspesifikk hastighetsmodell .	60
10.5	Tids- og hastighetsdata med GEH-verdier tilhørende figur 10.7, 10.8 og 10.9	63
10.6	Tids- og hastighetsdata med GEH-verdier tilhørende figur 10.10, 10.11 og 10.12	66
10.7	Tids- og hastighetsdata med GEH-verdier tilhørende figur 10.13	68
10.8	GEH-verdier for hastighetsmodell med tilpassede girskift og motordata . . .	70
10.9	GEH-verdier for hastighetsmodell med endret utnyttelsesgrad	72

1 Introduksjon

1.1 Bakgrunn

For å ha et forutsigbart trafikkbilde er det viktig å opprettholde den oppgitte kapasiteten på vegnettet. Et forutsigbart trafikkbilde gir økt sikkerhet, muliggjør effektiv ruteplanlegging og sørger for et pålitelig tilbud for brukerne.

I Norge er den foretrukne fraktmetoden ofte med lastebiler og trailere grunnet tilgjengelighet, hurtig levering, lavest kostnad og redusert behov for varelager. Industri og bosetning ligger spredt, og topografien i Norge gjenspeiles i vegnettet med liten radius i kurvene og store høydeforskjeller. Vegene i de mer rurale områdene av landet er stort sett tofeltsveger med varierende kvalitet. Disse krevende forholdene kan gjøre det vanskelig for tungt lastede kjøretøy å opprettholde fartsgrensen. Motstandskreftene i stigninger med høy gradient overgår i mange tilfeller de tilgjengelige motorkreftene, og hastigheten til kjøretøyet vil avta. Dette fører til redusert kapasitet på vegnettet, hvor tofeltsveger er spesielt utsatt.

Det er blitt utarbeidet en hastighetsmodell for tunge kjøretøy i stigning, som benytter flere statiske inngangsparametere sammen med fysiske lover for å beregne et teoretisk hastighetsprofil for tunge kjøretøy i stigninger. Hastighetsmodellen brukes blant annet som grunnlag av Statens vegvesen for dimensjonering av lengder på forbikjøringsfelt i stigninger. Det finnes muligheter for å optimalisere eksisterende hastighetsmodell ved bruk av erfaringstall fra forsøk med logging av reelle motordata, eksempelvis ved implementering av variabel motoreffekt og påvirkningen av girskift. En utvidet og mer optimalisert hastighetsmodell vil føre til mer realistisk teoretisk modellering som gir et bedre vurderingsgrunnlag for geometrisk utforming av veger, drivstoffberegninger, reisetidsberegninger, LCA-analyser og andre miljøberegninger.

Gjennom tidligere arbeider innen fagfeltet har det vist seg utfordrende å hente ut reelle motordata fra tunge kommersielle kjøretøy. Lastebilfabrikanter konstruerer kjøretøy som er langt mer restriktive med motordatatilgjengelighet under kjøring, sammenlignet med personbiler. Det er potensielt flere grunner til dette, men et sannsynlig ankepunkt er konkurranseaspektet. Lastebilfabrikanter leverer produkter som i større grad er i direkte konkurranse med hverandre, på et markant trangere marked enn personbilfabrikanter. De økonomiske konsekvensene for lastebilfabrikantene av å prestere bedre eller svakere på enkelte nøkkelparametere kan være enorme.

Området som går på motordatalogging, dataauthenting og databehandling fra tunge kjøretøy har behov for mer forskning. Det kommer stadig flere produsenter av motorloggerutstyr på markedet, som kan bidra til å muliggjøre dette. Et tett samarbeid med transportbransjen anses som en nøkkelfaktor da de er interessenter både ovenfor ny forskning for optimalisering og lastebilfabrikanter. Dette er nyttig for å skape åpenhet, felles målsetninger og avdramatisere utveksling av informasjon og opplysninger mellom fabrikanter og prosjekt.

1.2 Problemstilling

Formålet med masteroppgaven er å videreutvikle eksisterende hastighetsmodell for tunge kjøretøy i stigninger, utarbeidet av Børnes og Aakre. [Børnes og Aakre 2011] Dette skal gjøres ved å ta i bruk erfaringstall fra reelle forsøk med logging av motordata fra tunge kjøretøy. Erfaringstallene skal bidra til å gjøre hastighetsmodellen mer dynamisk og dermed projisere virkelige hastighetsprofil med en høyere grad av presisjon.

Med bakgrunn i målene er det blitt utformet to konkrete forskningsspørsmål som masteroppgaven skal gi svar på:

1. Hvordan kan loggerutstyr benyttes for å innhente motordata fra tunge kjøretøy?
2. Hvordan kan motordata anvendes for å videreutvikle en hastighetsmodell for tunge kjøretøy i stigning?

1.3 Prosjektoppbygging

Oppgaven er hovedsaklig bygget opp i kronologisk rekkefølge etter hvordan arbeidet har foregått gjennom prosjektet. I kapittel 2 presenteres grunnleggende sammenhenger for modellering av kjøretøy. Det er fokus på drivkrefter og motstandskrefter som er selve grunnlaget for å modellere et kjøretøy i bevegelse.

I kapittel 3 legges eksisterende arbeider frem. Den eksisterende hastighetsmodellen til Børnes og Aakre (2011) introduseres, som store deler av prosjektet er bygget rundt. Bruken av modellen i praksis er også lagt frem, i form av Statens vegvesens krav til forbikjøringsfelt i stigning. Videre følger et utvalg av tidligere forskning basert på prosjektets interesseområde, med fokus på modellering av tunge kjøretøy.

Kapittel 4 omhandler datakommunikasjon og datagang i tunge kjøretøy, som er nyttig å ha kjennskap til før dataloggingsforsøk. Loggerinstrumenter for motor- og posisjonsdata følger i kapittel 5, med verktøy for fremstilling og sammenstilling av loggede data i kapittel 6.

Loggerforsøkene som har blitt gjennomført i prosjektet er beskrevet i kapittel 7 og 8. I første omgang med lette kjøretøy i kapittel 7, mens kapittel 8 inneholder de ulike forsøkene som ble gjort på tunge kjøretøy. Relevante tekniske data og annen informasjon tilhørende de ulike lastebilmodellene er også inkludert i disse kapitlene.

I kapittel 10 fremlegges forslaget til en utvidelse av eksisterende hastighetsmodell. Nye tilnærminger til motbakkefremstilling og motstandskrefter er sentrale tema, før en modellspesifikk hastighetsmodell presenteres. Oppbyggingen av den nye modellen og utvidelsespunkter som inkludering av gir og variabel motoreffekt forklares. Deretter følger bruk av den nye modellen i praksis satt opp mot resultater fra reelle forsøk og eksisterende hastighetsmodell. Slutten av kapittelet tar for seg begrensninger i den nye modellen, etterfulgt av modelleringsaspekter innenfor både ny og eksisterende modell som kan være verdt å se nærmere på i fremtiden.

Helt til slutt i oppgaven er en oppsummering av prosjektet i sin helhet, med fokus på hva som er blitt gjort og kartlagt direkte inn mot prosjektets formål og forskningsspørsmål, etterfulgt av forslag til videre arbeid.

1.4 Prosjektutførelse

Hovedveileder for masteroppgaven og forprosjektet [Nygaard og Kogstad 2020] til masteroppgaven, har vært Arvid Aakre - professor ved institutt for bygg- og miljøteknikk, NTNU. Det har vært regelmessige møter, hvor diskusjon og avklaring rundt bestilling av utstyr, fremdriftsplaner og innholdsvurdering har vært sentrale tema. Avdelingsingeniør Jan Erik Molde, ved institutt for bygg- og miljøteknikk, NTNU, har vært delaktig i hele prosessen og bistått mye med å etablere kontakt med transportaktører, arrangere møter, samt være lastebilsjåfør under forsøk.

Prosjektgruppa har under masteroppgaven hatt en løpende dialog med CSS Electronics - produsenter av anvendt motorloggerutstyr, hvor diskusjon rundt ulike loggerinstrumenter, uthenting av data og potensielle utfordringer og feilkilder har vært tema. Ulike lastebil-fabrikanter har også blitt kontaktet sentralt for diskusjon og avklaring rundt datagang, datakommunikasjon og motorloggermuligheter i kjøretøyene.

Statens vegvesen har vært involvert underveis i masteroppgaven, hovedsaklig i forbindelse med vegdata. Det er blitt gitt strategiske tips til hvordan ulike data kan hentes ut fra NVDB, samt gitt tilgang til etterspurte data.

Under forprosjektet til masteroppgaven ble det gjort et litteraturstudium, med påfølgende utredning av "State of the Art". Det stod tidlig i forprosjektet klart for prosjektgruppa hva den påfølgende masteroppgaven skulle handle om. Som følge av dette ble det gjort et grundig arbeid med den teoretiske bakgrunnen til masteroppgaven allerede i forprosjektet. Deler av masteroppgaven er derfor tatt direkte eller delvis fra forprosjektet. Hvilke kapitler dette omfatter er fremstilt i tabell 1.1, hvor kapitlene i parentes indikerer at de er blitt utvidet under masteroppgaven.

Kapittel	Delkapittel
1	(1.1), 1.2
2	2, 2.1 - (2.5)
3	3.1, 3.2, 3.3.6, 3.3.7
4	4.1, (4.2)
5	5, 5.1, (5.2)
7	7.1, 7.3

Tabell 1.1: Kapitler i masteroppgaven som er dekket i forprosjektet (Nygaard og Kogstad 2020)

1.5 Forkortelser og symbolforklaringer

Tabell 1.2 og 1.3 gir en oversikt over relevante forkortelser og symboler med tilhørende enhet som omtales i oppgaven. De fleste av forkortelsene og symbolene forklares når de introduseres i oppgaven, men påfølgende tabeller kan være et nyttig oppslagsverktøy ved gjennomlesning.

Forkortelse	Beskrivelse
AASHTO	Association of State Highway and Transportation Officials
ASAM	Association for Standardisation of Automation and Measuring Systems
CAN	Controller Area Network
CSV	Comma-separated values
DBC	Database Container format
DBN	Dot bracket format
FMS	Fleet Management System
GEH	Geoffrey E. Havers
GUI	Graphical User Interface
IRI	International Roughness Index
ISO	International Organization for Standardization
LCA	Life Cycle Assessment
MF4/MDF4	Measurement Data format version 4
MPD	Mean Profile Depth
NTNU	Norges teknisk-naturvitenskapelige universitet
NVDB	Nasjonal vegdatabank
OBD	On-board diagnostics
PGN	Parameter Group Number
SAE	Society of Automotive Engineers
SPN	Suspect Parameter Number
TXT	Text format

Tabell 1.2: Forkortelser med beskrivelse

Symbol	Beskrivelse	Enhet
A	Frontareal	m^2
a	Akselerasjon	m/s^2
α	Stigning	rad
C_D	Dragkoeffisient	-
η	Virkningsgrad drivlinje	-
F	Tilgjengelig drivkraft	N
F_K	Total kjøremotstand	N
F_L	Luftmotstand	N
F_R	Rullemotstand	N
F_S	Stigningsmotstand	N
f_r	Rullefriksjonskoeffisient	-
g	Gravitasjonskonstant = 9,81	m/s^2
i	Uttevslingsforhold turtall/drivhjul	-
k_m	Akselerasjonskoeffisient for roterende masser	-
k_r	Hastighetskonstant for rullemotstandberegning	-
k_s	Hastighetskonstant for luftmotstandberegning	-
m	Kjøretøymasse	kg
n	Turtall	Omdreining/min
n_d	Rotasjonshastighet drivhjul	Omdreining/min
P	Motoreffekt	kW
r	Drivhjulradius	m
ρ	Luftens massetetthet	kg/m^3
τ	Dreiemoment	Nm
v	Kjøretøyhastighet	m/s
v_L	Vindhastighet mot kjøreretning	m/s

Tabell 1.3: Symbolforklaringer med tilhørende enhet

2 Modellering av tunge kjøretøy

Ved matematisk modellering av hastigheten til et objekt må de fysiske kreftene som har påvirkning kunne kvantifiseres og beskrives med tall. Grunnleggende fysiske formler for ulike fenomener nyttes for å gi et mest mulig realistisk bilde. For å modellere et kjøretøy i bevegelse kan de fysiske bidragene avgrenses til to overordnede hovedgrupper, drivkrefter som gir et positivt bidrag i bevegelsesretningen og motstandskrefter som gir et negativt bidrag i bevegelsesretningen. Motstandskreftene som virker på et kjøretøy omtales mer spesifikt som kjøremotstand.

Det er valgt å benytte den anerkjente kjøretøyhåndboka fra Robert Bosch [Robert Bosch GmbH 2007] for å beskrive drivkreftene til et kjøretøy. For kjøremotstand er det brukt flere ulike kilder ettersom det er blitt gjort forskning som vurderes til å ha relevans for prosjektet.

2.1 Motorkraft

For at et legeme skal kunne akselerere må summen av krefter som bidrar i fartsretningen være større enn kreftene som virker i mot. Dette vil også gjelde for et kjøretøy hvor motorenheten skaper krefter som blir overført til hjulene via drivsystemet.

Sammenhengen mellom den tilgjengelige kraften på drivhjula (F), motoreffekten (P), drivlinjas virkningsgrad (η) og farten til kjøretøyet (v) er presentert i likning (1), hvor det kommer frem at kraften vil avta ved økende hastighet.

$$F = \frac{P \cdot \eta}{v} = \frac{\tau \cdot i}{r} \cdot \eta \quad (1)$$

2.2 Hastighet og turtall

Hastigheten til et kjøretøy kan beskrives ved å benytte rotasjonsfarten til drivhjula (n_d), drivhjulradius (r) og turtallet til motoren (n). Sammenhengen mellom turtallet og rotasjonsfarten til drivhjula kan beskrives med et utvekslingsforhold for et bestemt gir, og kan uttrykkes ved: $i = \frac{n}{n_d}$. Høyere gir vil gi et større utvekslingsforhold som igjen vil gi høyere rotasjonshastighet ved et gitt turtall.

$$v = \frac{2\pi r \cdot n}{60 \cdot i} \quad (2)$$

2.3 Kjøretøydynamikk og drivkrefter

For kjøretøy blir ofte motoreffekten oppgitt i hestekrefter (hk), som kan konverteres til SI-enheten for effekt, watt (W). Størrelsen på effekten gjør det hensiktsmessig å benytte kilowatt (kW), hvor 1 hk tilsvarer 0,735 kW. Kombinasjon av likningen for krafta på drivhjulene (1) og likningen for hastigheten (2), gir at motoreffekten (P) er proporsjonal med produktet av dreiemomentet (τ) og turtallet (n), som er vist i likning (3). For å

konvertere turtall fra omdreininger per minutt til radianer per sekund benyttes koeffisienten $\frac{2\pi}{60}$.

$$P = \tau \cdot n \cdot \frac{2\pi}{60} \quad (3)$$

2.4 Akselerasjon

Akselerasjonen til et kjøretøy ved et gitt tidspunkt kan beregnes ved hjelp av Newtons andre lov, som sier at kraft er et produkt av masse og akselerasjon. Akselerasjonen til et kjøretøy vil være et resultat av størrelsen på drivkreftene (F), kjøremotstanden (F_K), massen til kjøretøyet (m) og kreftene som går med til å akselerere roterende masser. Formelen for akselerasjon er vist i likning (4). Ved roterende masser menes hjul, drivaksling og andre roterende elementer i systemet, og koeffisienten k_m beskriver betydningen av å akselerere disse elementene i forhold til kjøretøyet masse. For tunge kjøretøy vil differansen mellom den roterende massen og kjøretøyet masse være av en slik størrelse at $k_m \approx 1$.

$$a = \frac{F - F_K}{k_m \cdot m} \quad (4)$$

2.5 Kjøremotstand

Kjøremotstand er motstandskrefter på kjøretøyet som må overkommes for å skape fremdrift. Normen for studier som fokuserer på modellering av kjøretøy er å gjøre en forenkling som inkluderer de dominerende kvantifiserbare motstandskreftene, luftmotstand (F_L), rullemotstand (F_R) og stigningsmotstand (F_S). Summen av disse tre motstandskreftene utgjør den totale kjøremotstanden (F_K).

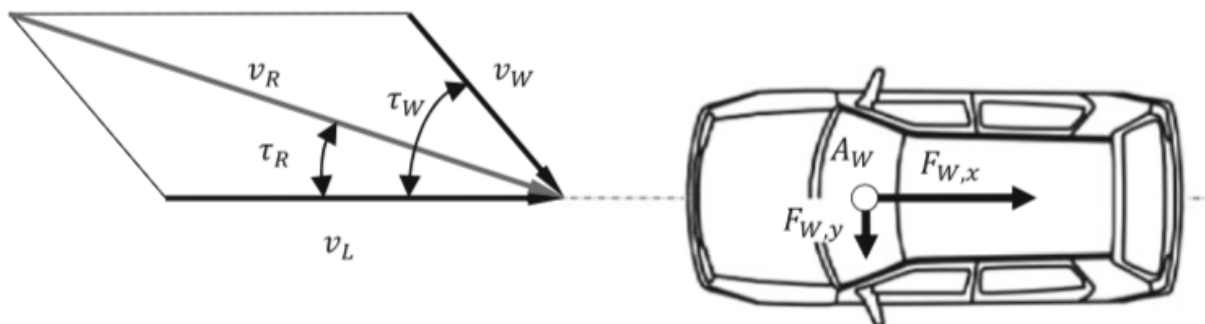
$$F_K = F_L + F_R + F_S \quad (5)$$

2.5.1 Luftmotstand

Luftmotstand er motstandskrefter som virker på legemer i bevegelse gjennom luft. Den totale luftmotstanden på kjøretøy kan deles inn i tre fysiske fenomener [Schramm, Hiller og Bardini 2014]:

- Motstand som følge av turbulens mot bakenden av kjøretøyet – ca. 85%
- Overflatemotstand som følge av skjærkrefter langs kjøretøyet – ca. 10%
- Indre motstand av luften som passerer gjennom kjøretøyet – ca. 5%

Når et kjøretøy er i bevegelse, kommer størsteparten av luftmotstanden fra turbulens, grunnet trykkforskjeller i luften foran og bak kjøretøyet. Ved å betrakte et todimensjonalt system kan den relative vindhastigheten beregnes ved hjelp av vinklene (τ_R) og (τ_W) som fremgår av figur 2.1.



Figur 2.1: Relative lufthastigheter mot kjøretøy [Schramm, Hiller og Bardini 2014]

For beregninger i kjøretøymodeller er det et ønske om å betrakte kreftene som virker i longitudinal retning. Motstandskreftene som kjøretøyet opplever grunnet luftmotstand kan derfor beskrives som

$$F_L = \frac{1}{2} \cdot \rho \cdot A \cdot C_D \cdot (v_0 + v_L)^2 \quad (6)$$

hvor ρ er luftens massetetthet, A er frontarealet til kjøretøyet og C_D er en dimensjonsløs dragkoeffisient. Dragkoeffisienten påvirkes av utformingen til objektet den beskriver, hvor en mer strømlinjeformet utforming gir en lavere tallverdi. Rekha mfl. (2001) fremlegger ulike typer kjøretøy med tilhørende dragkoeffisient. Et utvalg er gjengitt i tabell 2.1. v_0 og v_L beskriver den relative hastighetsforskjellen mellom henholdsvis den stillestående luften og kjøretøyet, og den ikke-stillestående luften og kjøretøyet. Motvind (v_L) i forhold til kjøreretningen vil være et positivt bidrag i formelen.

Kjøretøytype	C_D	Kilde
Semitrailer	0.70	Fitch (1994)
Søppelbil	0.95 - 1.05	SAEJ2188
Cattle hauler (Amerikansk snutedesign)	1.1	SAEJ2188
Kjøretøy med aerodynamisk utforming på tak	0.64	SAEJ2188

Tabell 2.1: Utvalg av kjøretøytyper med tilhørende dragkoeffisient, C_D [Rakha, Lucic mfl. 2001]

2.5.2 Rullemotstand

Ved å betrakte et legeme som *ikke* er frittrullende over et underlag, vil det dannes et høyere trykk i fremre del av legemet sammenlignet med bakre del. Det vil oppstå en longitudinal resultantkraft i kontaktpunktet mellom legemet og underlaget, som består av bidrag fra ulike fenomener. Plastisk deformasjon rundt kontaktpunktet, elastisk hysteresis og adhesjon er store bidragsytere til rullemotstand. [Ai mfl. 2011]

For et dekk som ruller over et asfaltdekke, vil parametere som har betydning for rullemotstand typisk være underlagets ruhet, aksellast, dekktrykk, dekktemperatur, hastighet, dekkmaterial- og utforming, og slippvinkel. [Gillespie 1992] Flertallet av modeller bruker

en tilnærming hvor rullemotstanden er et produkt av normalkraften mot underlaget og en rullefriksjonskoeffisient,

$$F_R = f_r \cdot W = f_r \cdot m \cdot g \cdot \cos(\alpha) \quad (7)$$

hvor f_r er rullefriksjonskoeffisienten, m er massen til kjøretøyet, g er gravitasjonskonstanten og α er vinkelen mellom en potensiell stigning i forhold til horisontalplanet, angitt i radianer. Ved transportinstituttet i Sverige er det blitt gjort empiriske forsøk med lastebil både med og uten henger for å bestemme rullefriksjonskoeffisienten (f_r). [Hammarström mfl. 2012] Parametere som inngår i modellen for lastebil er indeksverdien til dekkets overflateruhet (iri) i m/km, dekkets makrotekstur (mpd) i mm og lastebilens hastighet (v) i m/s. Formelen er presentert i ligning 8.

$$f_r = 0,00414 + 0,0000158 \cdot iri \cdot v + 0,00102 \cdot mpd \quad (8)$$

Gjennomsnittlig iri for det svenske vegnettet i studien er estimert til å være 2,4 m/km. Gjennomsnittlig mpd er estimert til å være 0,9 mm. Ved bruk av disse gjennomsnittsverdiene kan rullefriksjonskoeffisienten (f_r) varieres med hastigheten (v) i m/s. Med data fra Statens vegvesen kan også iri og mpd varieres for å gi en mer presis opptredende rullefriksjonskoeffisient for gitte strekninger i Norge.

En annen tilnærming til rullemotstand blir blant andre benyttet av Lan og Menendez (2003). For en lastebil på et asfaltert dekke kan den generelle rullemotstanden beskrives av ligningen

$$F_R = f_r \cdot W = (k_r + k_s \cdot v) \cdot m \cdot g \quad (9)$$

hvor k_r og k_s er konstanter som tar hensyn til påvirkningen hastighet har på rullemotstand. Hastigheten (v) oppgis i m/s. Formelen er en forenkling av de mange friksjonsfaktorene som spiller inn, som har ført til variasjoner på tvers av studier i tallfestingen av konstantene k_r og k_s . Lan og Menendez (2003) argumenterer for en tallverdi $k_r = 0,01$ og $k_s = 1/4470$ gir en akseptabel approksimasjon for lastebiler på asfalt i hastigheter opp til 128 km/t. [Lan og Menendez 2003]

Studiet av LaClair (2005) støtter opp under den generelle oppfatningen om at økt hastighet gir økt rullemotstand. Flere fenomener spiller inn på rullemotstanden ved økt hastighet, men på generell basis vil rullemotstanden øke som et resultat av økte sentrifugalkrefter. Økte sentrifugalkrefter fører til økt dekkradius på motsatt side av kontaktpunktet, mens deformasjon i kontaktpunktet forblir nært kontinuerlig under ulike hastigheter. Dette resulterer i at den relative forskyvningen under hver omdreining blir større for høyere hastigheter, som fører til et større energitap. [LaClair 2005]

2.5.3 Stigningsmotstand

Stigningsmotstand beskriver motstand som følge av endring i potensiell energi. Gravitasjonskraften trekker på alle objekter på jorda med en kraft som beskrives av Newtons andre lov,

$$F_S = m \cdot g \cdot \sin(\alpha) \quad (10)$$

hvor m er massen til objektet, g er jordas gravitasjonskonstant og α er vinkelen mellom underlaget og horisontalplanet, angitt i radianer. Stigningsmotstanden på et kjøretøy i motbakke vil derfor være tilsvarende.

3 Eksisterende arbeider

I kapittel 3 presenteres hastighetsmodellen til Børnes og Aakre (2011) som er modellen prosjektet har tatt utgangspunkt i når det kommer til å implementere ny forskning for utvidelse og optimalisering. Statens vegvesens krav til forbikjøringsfelt i stigning er også lagt frem, som delvis benytter Børnes og Aakres modell i praksis.

Tidligere forskning er presentert, med fokus på studier som er relevant for oppgaven. Dette er ment som en kartlegging av hva som er blitt gjort, hvordan forskningen er blitt gjort samt hva som kan gjøres for å ta forskningsområdet videre.

3.1 Børnes og Aakre (2011)

Formålet med modellen til Børnes og Aakre er å etablere en predikerende hastighetsmodell som kan brukes til å estimere reisetid for tunge kjøretøy som en funksjon av ulike kjøretøy-, veg- og trafikkparametere. Modellen fokuserer kun på vertikalkurvatur, drivkrefter og kjøremotstand, men det understrekes at langt flere faktorer vil spille inn på hastighetsprofilen i virkeligheten. [Børnes og Aakre 2011]

Modellen kalkulerer et teoretisk hastighetsprofil basert på inngangsdata som kan deles opp i tre kategorier. Disse kategoriene er kjøretøyparametere, stigningsbeskrivelse og kjøremotstand. I kjøretøyparametere inngår kjøretøyets masse, maksimal motorkraft, utnyttelsesgrad av motorkraft og starthastighet. I stigningsbeskrivelse inngår den prosentvise vertikale helningen og lengden, oppdelt i inntil 5 delstrekker. I kjøremotstand inngår rulle- og luftmotstand og stigningsmotstand. Rullemotstand beregnes ved bruk av kjøretøyets masse, jordens gravitasjonskonstant, den vertikale helningen og en friksjonskoeffisient. Luftmotstand beregnes ved bruk av luftens massetetthet, kjøretøyets frontareal, hastighet og dragkoeffisient, samt vindhastighet relativ til kjøretøyets bevegelsesretning. Stigningsmotstand beregnes ved bruk av kjøretøyets masse, jordas gravitasjonskonstant og den vertikale helningen. En oversikt over statiske inputparametere til modellen kan sees i tabell 3.1.

Parameter	Symbol	Enhet
Kjøretøy	m	kg
Maksimal effekt	P_{\max}	kW
Starthastighet	v_1	km/t
Makshastighet	v_{\max}	km/t
Maksimal akselerasjon	a_{\max}	m/s^2
Frontareal	A	m^2
Vindhastighet	v_0	m/s
Rullefriksjonskoeffisient	f_r	-
Dragkoeffisient	c_w	-
Luftens massetetthet	ρ	kg/m^3
Tyngdeakselerasjon	g	m/s^2

Tabell 3.1: Statistiske inputparametere for hastighetsmodell [Børnes og Aakre 2011]

Modellen er bygget i regnearkprogrammet Excel. Der beregnes hastigheten til kjøretøyet i tidsintervaller ut i fra kraftlikevekt mellom kjøretøyet drivkrefter og kjøremotstand, basert på inputparametere i tabell 3.1 og 3.2. Vertikalkurvatur for en ønsket stigning oppgis i form av lengde og helning, med tilhørende utnyttelsesgrad fra kjøretøyet motor. Utnyttelsesgraden forklarer et effekttap i motoren, som i hovedsak kommer av drivverksmotstand grunnet interaksjonen mellom motorens bevegelige deler. Segmentet med vertikalkurvatur og utnyttelsesgrad gjøres mer dynamisk ved muligheten for oppdeling i inntil 5 delstrekker. Dynamiske inputparametere for delstrekker kan sees i tabell 3.2.

Parameter	Enhet
Lengde på delstrekke	m
Helning på delstrekke	%
Utnyttelsesgrad til motor	%

Tabell 3.2: Dynamiske inputparametere for hastighetsmodell [Børnes og Aakre 2011]

Modellen er kalibrert og validert med reelle observasjoner fra feltundersøkelser. Et vegsegment på 70 km ble kjørt med ulike kombinasjoner av tunge kjøretøy, førere og last. På vegsegmentet var det flere stigninger av en slik grad at kjøretøyet terminale krabbhastighet ble oppnådd, hvor det er kraftlikevekt mellom driv- og motstandskreftene, dermed ingen akselerasjon. Et eksempel på inndata til modellen er gitt i figur 3.1, med tilhørende estimert fartsprofil i figur 3.2.

GRUNNLAGSDATA (gule (og grønne) felt kan endres): 0,8

Kommentar:

Modell

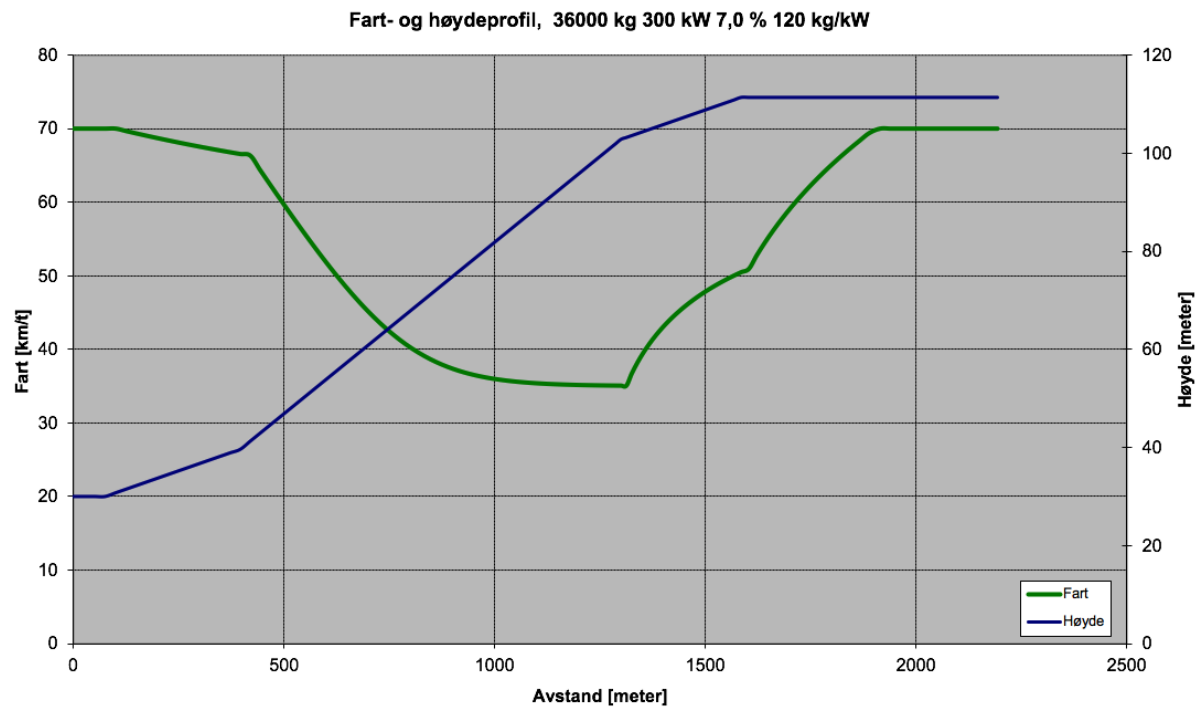
Tidsintervall	deltaT	1,30	sekund	Brukes for å formattere grafen
Tyngdeakselerasjon	g	10	m/s^2	
Tetthet luft	rho	1,20	kg/m^3	Ved vanlig temperatur og trykk
Vindhastighet	v0	0,0	m/s	(+ = motvind)
Rullemotstand	f	0,015		Typisk område 0.010 - 0.020
Luftmotstand	cw	0,60		Personbil 0.30-0.50, lastebil 0.50 - 0.70
Areal	A	8,0	m^2	Personbil ca 2 m^2, lastebil ca 8 m^2
Startavstand	x1	0,0	meter	For grafisk framstilling
Starthøyde	h1	30,0	meter	For grafisk framstilling
Starthastighet	v1	70	km/t	
Maxhastighet	v_max	70	km/t	
Max akselerasjon	a_max	3,0	m/s^2	KLADD - ulike mål for effekt
Masse	m	36000	kg	Dreiemoment
Maks effekt	P_max	300,0	kW	Turtall
Tilsvare i HK		408,0	HK	Effekt
Masse / effekt forhold		120,0	kg/kW	Tilsvare i HK
		88,2	kg/HK	
Effekt / masse forhold		8,33	W/kg	Effekt i HK
		11,33	HK/tonn	Tilsvare i kW
Tidsforbruk	Lett	2193	meter	70,0 km/t
	Tung	2193	meter	50,6 km/t
				112,8 sek
				156,0 sek
				43,2 sek

Angi utnyttelse av effekten for hver delstrekning i tabellen under

Delstrekninger:

	stigning S prosent	lengde L meter	fra meter	til meter	høyde H meter	utnyttelse grad u prosent	utnyttet effekt P Watt
1	0,0	100	0	100	0,0	95	285000
2	3,0	300	100	400	9,0	95	285000
3	7,0	900	400	1300	63,0	100	300000
4	3,0	300	1300	1600	9,0	95	285000
5	0,0	100	1600	1700	0,0	95	285000
Sum		1700			81,0		

Figur 3.1: Eksempel med inndata til hastighetsmodell [Børnes og Aakre 2011]



Figur 3.2: Eksempel med predikert hastighetsprofil fra hastighetsmodell [Børnes og Aakre 2011]

3.2 Statens vegvesens krav til forbikjøringsfelt i stigning

Statens vegvesens håndbok N100 Veg- og gateutforming stiller krav til forbikjøringsfelt i stigning. Hensikten er å ivareta et jevnt fartsnivå, derfor skal forbikjøringsfelt vurderes hvor hastighetsdifferansen mellom tunge og lette kjøretøy blir for stor. [Vegdirektoratet, Statens vegvesen 2019a] Hastighetsdifferansen som kategoriseres som for stor og dermed utløser et behov for forbikjøringsfelt, er satt til 15 km/t. Forbikjøringsfeltet bør avsluttes når fartsdifferansen er nådd 10 km/t. Dette gjelder for vegstrekker hvor antall tunge kjøretøy per døgn er minst 400. Det presiseres at i stigninger med positiv helning skal det benyttes en regnearkmodell for beregning av lengden på forbikjøringsfeltet.

Regnearkmodellen som det refereres til i håndbok N100 benytter fysiske lover for å beregne hastigheten til et tungt kjøretøy i motbakke, for så å anslå start og slutt punkt for et forbikjøringsfelt basert på hastighetsdifferansen mellom det tunge kjøretøyet og fartsgrensen. Hastigheten beregnes i tidsintervaller ut i fra kraftlikevekt mellom kjøretøyet drivkrefter og kjøremotstand. Stigningen som det skal beregnes for kan oppdeles i inntil 10 delstrekker. Modellen har i prinsippet lik oppbygning som modellen til Børnes og Aakre som er beskrevet i kapittel 3.1 En viktig forskjell mellom modellene er de ulike inngangsparameterene som kan velges. I modellen beskrevet i håndbok N100 er en rekke parametere fastsatt ut i fra dimensjonerende forhold. Dimensjonerende verdier for tunge kjøretøy (vogntog) i modellen finnes i håndbok V120 Premisser for geometrisk utforming av veger [Vegdirektoratet, Statens vegvesen 2019b], og er gjengitt i tabell 3.3. Et eksempel med bruk av modellen er lagt ved i figur 3.3.

Parameter	Enhet	Verdi
Rullefriksjonskoeffisient	-	0,015
Dragkoeffisient	-	0,6
Frontareal	m ²	8,0
Masse	kg	40 000
Motoreffekt	kW	360
Utnyttelsesgrad	%	95

Tabell 3.3: Dimensjonerende verdier som inngår i beregning av akselerasjon for vogntog [Vegdirektoratet, Statens vegvesen 2019b]

FORBIKJØRINGSFELT I STIGNING

Regneark for beregning av forbikjøringsfelt i stigninger

Versjon 1.0 / 2012-12-12

Forenklet stigningsforløp:

Stigningsforløpet forenkles til opptil 9 delstrekninger, iht beskrivelse i håndbok 265. Delstrekningenes stigning og lengde settes i tabellen.

	Stigning [%]	Lengde [m]
L1	0,0	400
L2	3,5	400
L3	7,0	600
L4	3,5	400
L5	0,0	1000
L6		
L7		
L8		
L9		

Fartsgrense

Vf 80 km/t

Fartsgrensen på veien

Startfart (valgfritt)

V1 km/t

Hvis startfarten er ulik fartsgrensen, tast inn verdi.

Kritisk fartsdifferanse

$\Delta V_{k,start}$ 15 km/t

Jmfr. Håndbok 265

$\Delta V_{k,slutt}$ 10 km/t

Startprofil (valgfritt)

x1 0,0 m

Profilnummer ved starten av L1.

Starthøyde (valgfritt)

h1 633,2 m.o.h.

Høyde ved starten av L1.

Antall sekunder

T 200,0 sek.

Modellens beregningstidsrom

Punkt A der farten < 65 km/t inntreffer etter:

887 m

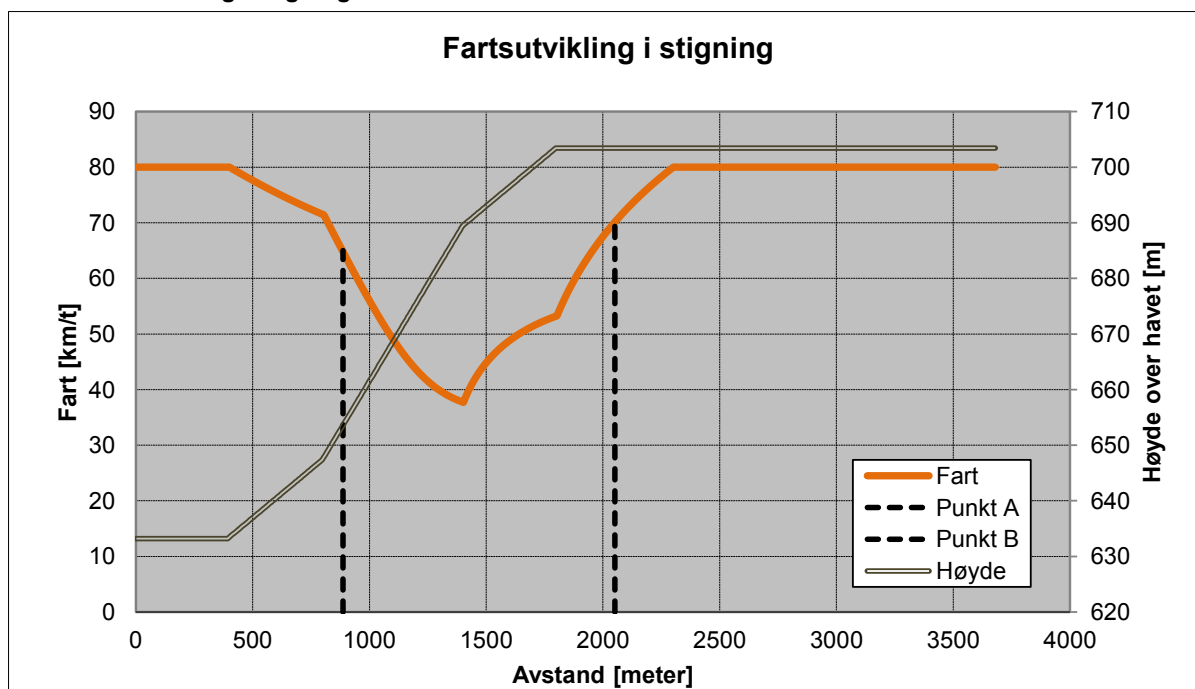
Punkt B der farten ≥ 70 km/t inntreffer etter:

2051 m

Forbikjøringsfeltets lengde, minimum:

1164 m

Graf: Fartsutvikling i stigning



Figur 3.3: Eksempel med regnearkmodell for forbikjøringsfelt i stigning [Vegdirektoratet, Statens vegvesen 2019a]

3.3 Tidligere forskning

Proessen og metodikken rundt litteratursøk og kartlegging av tidligere forskning startet under forprosjektet til masteroppgaven. [Nygaard og Kogstad 2020] Blant målene var å oppnå en forståelse for søkemetodikk og kildekritikk. Kildene er en kombinasjon av forslag fra veileder, avgrensede søk i søkemotorer og snowballing. For mer detaljert fremgangsmåte for funn og vurdering av eksisterende litteratur henvises det til forprosjektet til masteroppgaven.

3.3.1 Archilla og De Cieza (1996)

I studiet fra 1996 av Archilla og Cieza ble det samlet inn større mengder hastighetsdata fra tunge kjøretøy på Argentinske motorveger, som ble brukt for å utvikle en teoretisk hastighetsmodell basert på kjøretøyets vekt/kraft-forhold. [Archilla og De Cieza 1996]

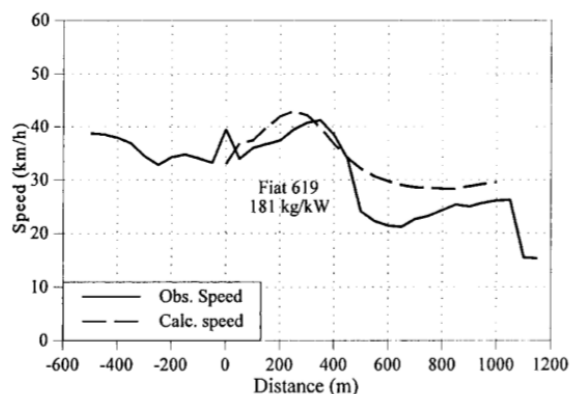
Totalt ble det registrert i underkant av 3000 kjøretøy av alle typer, med tre punkthastigheter i tre forskjellige stigninger med ulik helning. 31,4 % av registreringene var lastebiler. I de to bratteste stigningene på 4,2 % og 6,2 %, ble krabbehastighet oppnådd og registrert på de frie tunge kjøretøyene. Ved hjelp av skiltavlesning ble kjøretøymodell med motorkraft sammen med registrert vekt fra vektstasjoner funnet i databaser i etterkant.

Den teoretiske hastighetsmodellen som ble utformet bygger i stor grad på de samme grunnleggende fysiske formlene for modellering av kjøretøy som er presentert i kapittel 2. Den direkte påvirkningen av girvalg på hastighetsprofilen er ikke inkludert i den teoretiske modellen, ettersom dette ville gjort modellen langt mer komplisert. For å inkludere motstanden som følge av girskift, ble det gjort en regresjonsanalyse av observerte krabbehastigheter for å finne en konstant motstandsfaktor. Ytterligere hastighetsregistreringer med høyere loggerfrekvens av lastebiler i de to bratteste stigningene ble utført for å validere modellen.

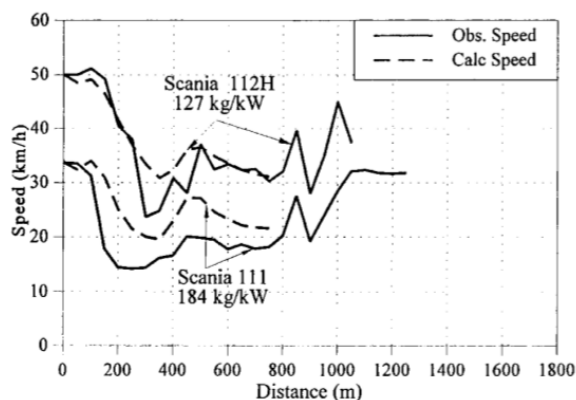
I figur 3.4 er to ytterpunkter av registrert hastighet i stigningen på 4,2 % satt opp mot modellen. I figur 3.4 (a) følger modellert hastighet den registrerte hastigheten relativt tett. I figur 3.4 (b) er det større forskjell mellom modellert og registrert hastighet i starten av stigningen, før forskjellen reduseres. Studiet argumenterer med at i figur 3.4 (b) har sjåføren valgt et gir med mål om å holde konstant hastighet gjennom hele stigningen. Modellen er bygget opp på kraftlikevektsligninger som forutsetter at kjøretøyet holder høyest mulig hastighet, derfor blir utslaget større.

Figur 3.5 (a) og (b) viser hastighetsprofilene til fire lastebiler i stigningen på 6,2 %. I motsetning til stigningen på 4,2 %, har lastebilene i denne stigningen mindre frihet til å velge hvordan stigningen skal angripes. Det ser tilsynelatende ut til at maksimal tilgjengelig motorkraft blir utnyttet, som gir en mer nøyaktig modellert hastighet.

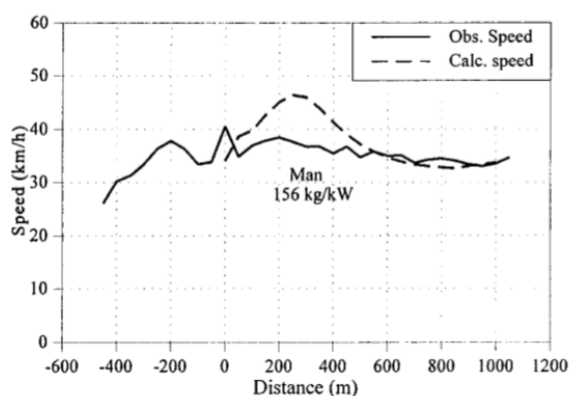
Studiet konkluderer med at den teoretiske hastighetsmodellen totalt forklarer rundt 80 % av observerte hastighetsvariasjoner. Nøyaktigheten til modellen er høyere for brattere stigninger, mens for mindre bratte stigninger gjør sjåførens kjørestil med girvalg større utslag på hastighetsprofilen.



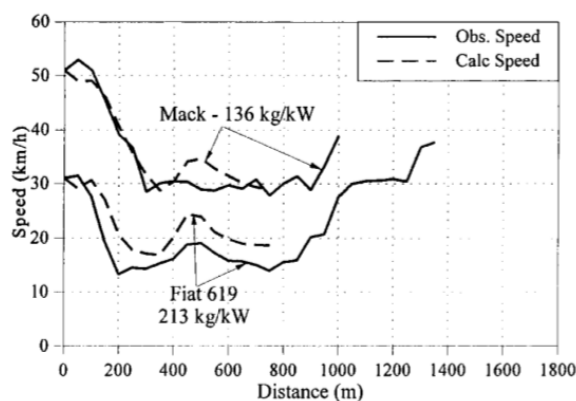
(a)



(a)



(b)



(b)

Figur 3.4: Registrert mot modellert hastighet i stigning på 4,2 % [Archilla og De Cieza 1996]

Figur 3.5: Registrert mot modellert hastighet i stigning på 6,2 % [Archilla og De Cieza 1996]

3.3.2 Bester (2000)

Bester utviklet i 2000 en matematisk hastighetsmodell for lastebiler, med antakelse om at lastebilens akselerasjon er en lineær funksjon av hastigheten. Hastighetsmodellen er utviklet spesifikt for motorvegsegmenter hvor vertikalkurvaturen er den begrensende faktoren for lastebilenes hastighet. [Bester 2000]

Hastighetsmodellen er i stor grad bygget opp av de samme fysiske formlene som er presentert i kapittel 2, men nedskaleres parameterne knyttet til motorytelse, kraft/vektforhold og motstandskrefter til kun to konstanter, α og β . Formelen for akselerasjon blir da som beskrevet i ligning 11, hvor V er hastigheten, g er tyngdeakselerasjonen, G er stigningsgradienten, og M/M_e er en funksjon som tar hensyn til total effektiv masse som må akselereres, i form av statisk og roterende masse.

$$a = \alpha - \beta \cdot V - g \cdot G \cdot M/M_e \quad (11)$$

α og β kan bestemmes for individuelle lastbiler og grupper av lastebiler. For individuell bestemmelse gjennomføres en akselerasjonstest, hvor tiden (t) det tar å nå en viss hastighet noteres. Maksimal hastighet for kjøretøyet vil tilsvare α/β i en situasjon hvor det verken

er stigning eller akselerasjon. Dette kommer frem av ligning 11. Ved å bruke tall fra disse registreringene, kan α og β bestemmes ved bruk av ligning 12.

$$V = \alpha/\beta(1 - e^{-\beta t}) + V_0 e^{-\beta t} \quad (12)$$

For bestemmelse av α og β for en større gruppe med lastebiler registreres hastighetsprofiler gjennom tre til fire punkt langs et strekke med kjent vertikal stigning. Ved et lengre strekke hvor lastebilene oppnår krabbhastighet kan ligning 11 brukes direkte til å bestemme α og β , ettersom akselerasjonen er null, og G og V er kjent.

Besters hastighetsmodell er en forenkling av lastebilers oppførsel i motbakke, men modellens simplisitet gjør den enkel å anvende, samt at den kan kalibreres for et gitt utvalg av lastebiler.

3.3.3 Rakha og Lucic (2002)

Rakha og Lucic bygget i 2002 videre på egne arbeider ved å implementere variabel motoreffekt i sin dynamiske modell for predikering av akselerasjon og hastighet til kjøretøy. [Rakha og Lucic 2002] Modellen regner ut akselerasjon i tidsintervaller med en gitt steglengde basert på resultantkraften i hvert tidsintervall. Resultantkraften beregnes ut i fra drivkrefter i form av tilgjengelig motoreffekt og motstandskrefter som består av stigningsmotstand, luftmotstand og rullemotstand. Formelen for akselerasjon er beskrevet i ligning (13), hvor F er drivkrefter, R er motstandskrefter og M er kjøretøyets masse.

$$a(t_i) = \frac{F(t_i) - R(t_i)}{M} \quad (13)$$

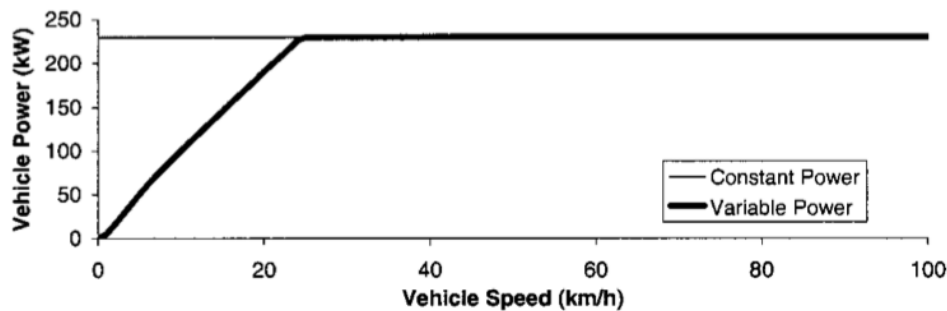
Studiet argumenterer med at det vil være uriktig å anvende konstant motoreffekt langs hele hastighetsspekteret for tunge kjøretøy, ettersom det gir urealistisk høye akselerasjons-egenskaper ved lave hastigheter. Ved lave hastigheter vil det være hyppigere girskift, hvor motoren ikke vil produsere maksimal tilgjengelig effekt. Rakha og Lucic løser dette ved å modellere kjøretøyets drivkraft som en lineær funksjon av hastighet fra kjøretøyet er stillestående til hastigheten hvor maksimal motoreffekt nås. Matematisk blir dette gjort ved å tillegge en faktor β i formelen for tilgjengelig drivkraft (F_t). Dette er vist i ligning (14), (15), (16) og er illustrert i figur 3.6.

$$F_t = 3600 \cdot \beta \cdot \eta \cdot \frac{P}{v} \quad (14)$$

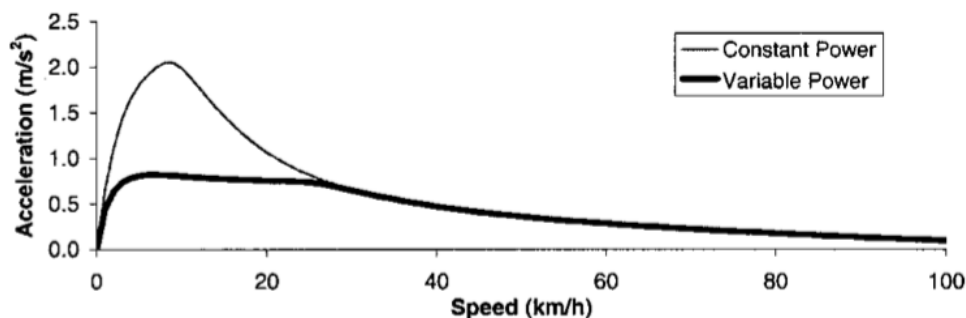
$$\beta = \frac{1}{v_0} \left[1 + \min(v, v_0) \left(1 - \frac{1}{v_0} \right) \right] \quad (15)$$

$$v_0 = 1,164 \cdot w^{-0,75} \quad (16)$$

P er maksimal motoreffekt, η er motorens virkningsgrad, v og v_0 er punkthastighet og starthastighet. w beskriver vekt/kraft-forholdet til kjøretøyet. Innvirkningen variabel motoreffekt har på maksimal modellert akselerasjon er illustrert i figur 3.7.



Figur 3.6: Variabel drivkraft som en funksjon av hastighet [Rakha og Lucic 2002]



Figur 3.7: Modellert akselerasjon ved implementering av variabel motoreffekt [Rakha og Lucic 2002]

Modellen ble kalibrert ved feltforsøk med fire ulike manuellgirede lastebiler med ulike vekt/kraft-forhold. Hver enkelt lastebil ble testet med minst 10 ulike vektkonfigurasjoner.

3.3.4 Lan og Menendez (2003)

Lan og Menendez la i 2003 frem en fremgangsmåte og formel for å beregne hastighetsprofiler til lastebiler, basert på dynamiske, kinematiske og operasjonelle egenskaper i motbakke. [Lan og Menendez 2003] Bakgrunnen for studiet er amerikanske AASHTOs (Association of State Highway and Transportation Officials) kriterier for etablering av krabbefelt i stigninger. AASHTO anbefaler bruk av et mer representativt vekt/kraft-forholdstall eller andre approksimasjoner enn standarden på 120 kg/kW på vegsegmenter hvor kjøretøyparken for tunge lastebiler har signifikant større aksellast i forhold til standardkjøretøyet med tilhørende designgrafer.

Grunnlaget for den matematiske modellen som blir utviklet er kraftlikevekt mellom drivkrefter og motstandskrefter, som brukes for å beregne akselerasjon. Modellen anvender Besters (2000) antakelser for å ta hensyn til effektiv masse og en tilnærming til rulle-motstand hvor motstandskraften varierer med hastigheten. Motstandsfaktor som følge av girskift blir vurdert til noe høyere enn anbefalt av Archilla og De Cieza (1996), med argumentet om at lastebiler oftere, også under lave hastigheter, benytter høyere gir med høyere overføringsmotstand.

Med hastighetsmodellen kan det gjøres vurderinger ut i fra resultater med lastebiler som fraviker standard vekt/kraft-forholdstall. Modellen klarer også å beregne over et større

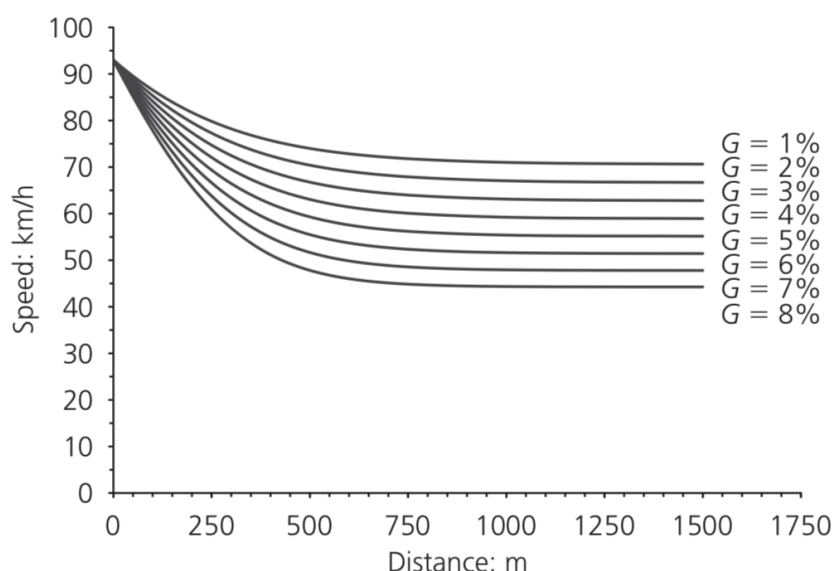
hastighetsspekter enn AASHTOs designgrafer kan tilby.

Lan og Menendez presenterer ved hjelp av modellen flere tabeller med kritisk lengde som funksjon av hastighetsøkning eller hastighetsreduksjon. Dette blir gjort for ulike hastigheter og stigningsgrader i oppover- og nedoverbakker. Resultatene er teoretiske og ikke validert i form av feltforsøk, men legges frem som et nyttig designsupplement.

3.3.5 Arellano, Echaveguren og Vargas-Tejeda (2014)

Det chilenske studiet fra 2014 ser på hastighetsprofiler til lastebiler i kortere motbakker. Studiet utforsker oppfatningen om at hastighetsprofilen til lastebiler i kortere motbakker vil være ulikt hastighetsprofilen som presenteres i tidligere modeller. I kortere motbakker vil det være tilgjengelige drivkrefter som tillater lastebilen å akselerere og dermed øke hastigheten i den siste tredjedelen av motbakken. [Arellano, Echaveguren og Vargas-Tejeda 2015]

I studiet blir Besters (2000) hastighetsmodell kalibrert for korte motbakker ved hjelp av feltforsøk med posisjonsloggerutstyr. Totalt ble det gjennomført 24 registreringer med lastebiler med ulikt vekt/kraft-forholdstall, i 24 ulike stigninger. Lavest registrert hastighet fra forsøkene var i snitt 15 % høyere enn teoretisk krabbehastighet beregnet med tradisjonell formel for kraftlikevekt. 13 av registreringene ble gjort med lastebiler med vekt/kraft-forholdstall i intervallet 95-160 kg/kW. Denne gruppen av lastebiler ble ansett som mest representativ for lastebilparken i Chile og tall fra disse registreringene ble brukt for å bestemme α og β i Besters modell. Kalibrerte hastighetsprofiler i ulike stigninger for lastebiler med et vekt/kraftforholdstall på 100-160 kg/kW og starthastighet på 70-93 km/t er illustrert i figur 3.8. Modellen er gjeldende for stigningslengder opp til 1500 meter.



Figur 3.8: Kalibrerte hastighetsprofiler for kjøretøy i motbakke < 1500 m, 100-160 kg/kW, 70-93 km/t [Arellano, Echaveguren og Vargas-Tejeda 2015]

Det presiseres at metoden som er brukt for kalibrering gjør modellen gjeldende for et lite nettverk av kjøretøy uten krav til omfattende detaljerte flåtedata, som gjør den enkel i bruk. Det er i studiet ikke tatt hensyn til varierende stigningsgrader i starten og slutten av motbakkene og effekten dette har på tilgjengelig motorkraft og akselerasjon.

3.3.6 Opstad (2015)

Erlend Opstad leverte i 2015 en erfaringsbasert masteroppgave ved NTNU, som så på behov for forbikjøringsfelt i stigning på Europaveg 39 Ørskogfjellet. [Opstad 2015] Størsteparten av oppgaven omhandler innsamling av reelle trafikkdata fra tunge kjøretøy i stigninger, samt tolkning og vurdering av innhentet data. Modellen til Børnes og Aakre blir mot slutten av oppgaven i et kort segment benyttet for å vurdere sammenhengen mellom teoretisk og observerte hastighetsprofil for tunge kjøretøy i stigninger. I den teoretiske hastighetsmodellen ble det brukt et vekt/kraft-forhold på 122,4 kg/kW.

Opstad påpeker at den teoretiske hastighetsmodellen danner et hastighetsprofil som er gjenkjennelig for kjøretøygruppe 24 og 25. (Kjøretøy som er lengre eller lik 12,5 meter) Det presiseres at andel tunge kjøretøy som er i gruppe 24 og 25 av den totale gruppen som omfatter tunge kjøretøy, (kjøretøy som er lengre eller lik 5,6 meter) er svært liten.

3.3.7 Nesdal (2019)

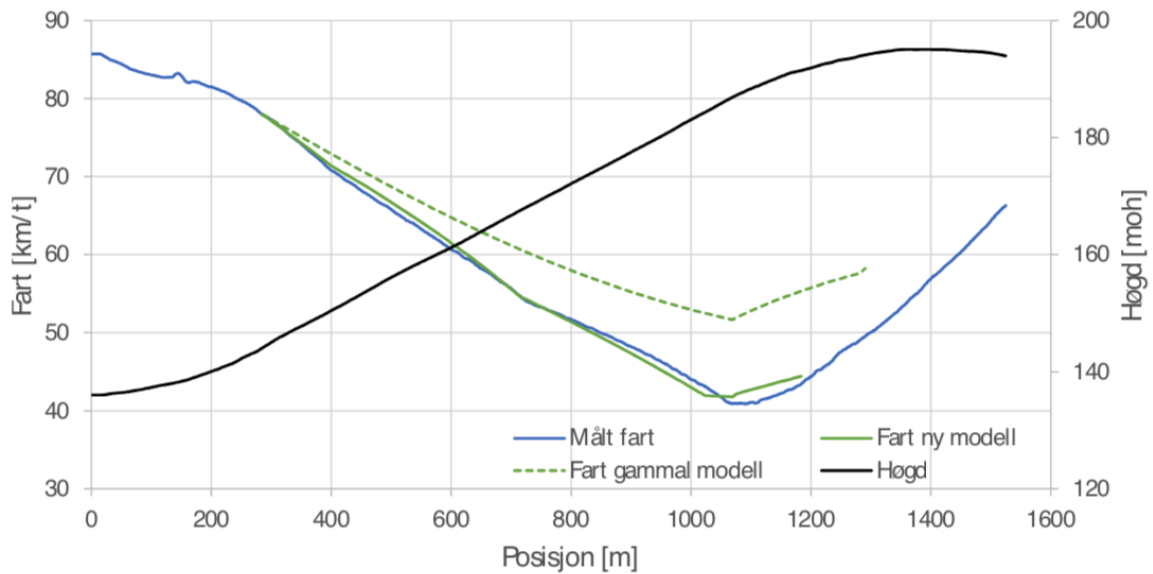
Andreas Nesdal leverte i 2019 en masteroppgave ved NTNU, som så på modellering av fartsprofil til tunge kjøretøy i stigning. [Nesdal 2019] Nesdal jobbet direkte mot Børnes og Aakres hastighetsmodell, med datainnsamling og videreutvikling.

Et av målene til Nesdal var å logge motordata og posisjonsdata fra feltforsøk med tunge kjøretøy. Dette skulle så brukes til å videreutvikle eksisterende hastighetsmodell, ved å modellere motoreffekten mer nøyaktig.

Nesdal erfarte i løpet av masteroppgaven at det viste seg utfordrende å hente ut motordata fra tunge kjøretøy. Grensesnitt mellom kjøretøykomponenter og loggerenheter, ulik fysisk utforming for informasjonsflyt i kjøretøyene, samt datakoding av eldre standard var blant utfordringene Nesdal erfarte. For Nesdal ble utfordringene med å hente ut motordata av en slik grad at det ble vurdert som urealistisk å kunne finne gode tekniske løsninger innen gjeldende tidsramme. Løsningen ble å filme instrumentpanelet og å logge posisjonsdata. Dette ble gjort under en forsøksstur med en lastebil med totalvekt på 50 tonn.

Med resultatene fra forsøkssturen utarbeidet Nesdal en funksjon for effekt som varierer med hastigheten til kjøretøyet. Nesdals funksjon for effekt er basert på den spesifikke momentkurven for lastebilen som ble benyttet i forsøket, hvor det er en lineær sammenheng mellom motoreffekt og produktet av dreiemoment og turtall. Momentkurven i kombinasjon med avlest hastighet, gir og turtall danner et grunnlag for å kunne variere motoreffekt med hastighet.

Nesdal gjennomførte modellberegninger for en utvalgt stigning fra forsøkssturen. Det ble gjennomført beregninger med den eksisterende hastighetsmodellen, deretter ved å inkludere variabel motoreffekt. For den nye modellen som inkluderer variabel motoreffekt ble friksjonsparameteren for rullestand nedskalert for å få en bedre kurvetilpasning. Resultatet kan sees i figur 3.9.



Figur 3.9: Implementering av variabel motoreffekt i hastighetsmodell [Nesdal 2019]

Det er åpenhet i masteroppgaven om at datagrunnlaget bak forskningen er mangelfullt. Modellen er basert på data fra en utvalgt stigning under én enkelt forsøktur, som gjør at resultatet ikke kan brukes til å etablere sammenhenger og fellestrekk for et større antall stigninger eller kjøretøy. Det presiseres at området trenger mer forskning, også med andre fokusområder enn Nesdal selv har hatt.

3.3.8 Xin mfl. (2020)

Cai Xin, Zhong Yi, Zhao Yong og Mao Yan publiserte i 2020 et studie som så på optimalisering av designparametere for longitudinale stigninger basert på kjøretøydynamikk for lastebiler. [Xin mfl. 2020] I studiet fokuseres det på lastebiler i motbakke som gir kapasitetsutfordringer, samt lastebiler i nedoverbakke som kan føre til varmgang i bremsene som gir nedsatt bremsekapasitet.

Programmet TruckSim ble brukt til å designe fire ulike lastebiler med ulike vekt/kraft-forhold, antall akslinger, antall gir og dekksett som ble simulert med ulike starthastigheter over ulike stigningsgrader. Simuleringene ble kalibrert ved bruk av en reell lastebil under feltforsøk. Av kalibreringen i motbakke kom det frem at den relative feilen mellom simulert stabil krabbestastighet og reell stabil krabbestastighet var på 7,24 %, hvor den reelle gjennomsnittlige hastigheten var lavere enn den simulerte.

Basert på kjøretøydynamikk og trafikksammensetning i Kina, blir det i studiet anbefalt en maksimal lengde på stigninger slik at hastighetsreduksjonen mellom starthastighet og stabil krabbestastighet er mindre enn 20 km/t.

3.4 Oppsummering

Samtlige tidligere arbeider som er gjengitt i kapittel 3.3 med unntak av Opstad (2015), ser på en form for hastighetsprofil for tunge kjøretøy i stigninger. En dynamisk tilnærming med beskrivelse av drivkrefter og motstandskrefter for beregning av akselerasjon og hastighet er gjeldende, unntaket er Bester (2000) som forenkler dynamikken ned til en mer rendyrket kinematisk ligning ved bruk av faktorene α og β .

Flere studier foreslår en hastighetsmodell og bruker reelle feltforsøk for å validere modellen. (Archilla og De Cieza (1996), Rakha og Lucic (2002), Arellano, Echaveguren og Vargas-Tajeda (2014), Nesdal (2019), Xin mfl. (2020)) Resterende legger frem en ren teoretisk modell uten direkte validering fra feltforsøk. (Bester (2000), Lan og Menendez (2003)) Både Rakha og Lucic (2002) og Nesdal (2019) har implementert en form for variabel motoreffekt i sine modeller.

Mange av studiene adresserer kompleksiteten ved å utvikle en teoretisk hastighetsmodell, hvor motorkarakteristikk, virkemåte og førervalg i realiteten har mye å si for hastighetsprofilene. Dette blir likevel sett bort i fra eller grovt forenklet da det ville ført til langt mer kompliserte modeller. Dette tyder på at det er behov for forskning på hvordan nevnte parametere påvirker hastighetsprofilene.

Av arbeidene som er gjennomgått er det kun Nesdal (2019) som har forsøkt å implementere motordata for å videreutvikle en hastighetsmodell. I studiet var det tenkt å registrere motordata med loggerutstyr, men dette ble til slutt kun gjort manuelt under én forsøksstur. Utfordringen med å implementere erfaringsdata fra elektronisk motorlogging i en hastighetsmodell for tunge kjøretøy i motbakke eksisterer ennå per dags dato.

4 Datakommunikasjon i tunge kjøretøy

I kapittel 4 er viktige kommunikasjonssystemer i tunge kjøretøy presentert. CAN-protokollen og SAE J1939-standarden legges frem, etterfulgt av OBD - On-board diagnostics og FMS - Fleet Management System. Ved direkte logging og behandling av data fra tunge kjøretøy, er det viktig å ha forståelse for hvordan dataen er bygget opp, datagang og tilgjengelighet.

4.1 CAN-protokollen

De aller fleste kjøretøy i dag benytter CAN-protokollen (Controller Area Network) som basis for å kommunisere mellom motorstyreenheter som eksempelvis airbag, ABS-system eller temperaturmålere. Protokollen ble utviklet og publisert av Bosch i 1986. I 1993 ble den oppført i ISO-standarden (International Organization for Standardization): ISO11898. [Prasad, Tang og Luo 2019]

CAN består fysisk av en to-tråds krets (CAN-high og CAN-low), terminert av en resistor i begge ender med tilkobling til mottakerene i de ulike motorstyreenhetene. Dette gjør kommunikasjon mellom ulike elementer mulig uten at de nødvendigvis trenger å være fysisk tilkoblet hverandre. Systemet er konstruert slik at det kan skille mellom dominante og recessive bit i meldinger fra ulike motorstyreenheter. En konsekvens av dette er eliminering av forsinkelse for meldinger med høy prioritet ved kollisjon med andre meldinger, ettersom meldingen med den dominante bit-en vil bli prioritert. Eksempelvis vil systemet prioritere en melding med informasjon om overoppheting i motoren kontra en melding om temperaturendring utendørs.

4.2 SAE J1939-standarden

I de fleste kommersielle tunge kjøretøy er det overordnede standardiserte programspråket SAE J1939-standarden, definert av SAE. (Society of Automotive Engineers) Fra et dataloggingsperspektiv fungerer J1939 som et tolkningslag over CAN, med et sett av standardiserte meldinger og transformasjonsregler som fungerer på et bredt spekter av tunge kjøretøy, i motsetning til lette personkjøretøy hvor de ulike fabrikantene benytter egne protokoller. SAE J1939-standarden spesifiserer i tillegg hvordan data kan konverteres til menneskelig lesbar form.

SAE J1939 ble tatt i bruk i år 2000, og har erstattet de tidligere standardene SAE J1708 og SAE J1587. Hastigheten var 250 kbit/s frem til 2015-2016, hvorpå hastigheten ble doblet til 500 kbit/s ettersom stadig flere motorstyreenheter i kjøretøyene krever større kapasitet. J1939 bruker i dag CAN 2.0B – et heksadesimalt tallformat hvor meldingene består av en identifiseringspakke på 29 bit, etterfulgt av en datapakke på 64 bit = 8 byte med rå CAN-data. J1939-meldingene blir identifisert ved hjelp av PGN (Parameter Group Number), som igjen inndeles i tre avgrensede identifiseringsenheter. [Prasad, Tang og Luo 2019] Av den totale identifiseringspakken på 29 bit består 18 bit av PGN, 3 bit er prioritering og 8 bit er kildeadresse. Et eksempel på en parametergruppe er motortemperatur, som blant annet omfatter drivstofftemperatur, oljetemperatur og kjølevæsketemperatur. Disse enkelte motorstyreenhetene som inngår i parametergruppen og gir data til datapakkedelen av J1939-meldingene er også nummererte, i form av SPN (Suspect Parameter Number).

Et eksempel på en PGN med tilhørende SPN-er er fremstilt i tabell 4.1 og 4.2. [Copperhill technologies 2020]

PGN 65262 - Engine temperature	
Transmission rate	1 sec
Data length	8 bytes
Default priority	6
PG Number	65262 (FEEE _{HEX})

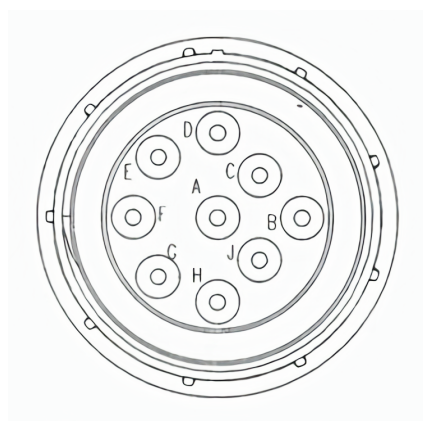
Tabell 4.1: PGN-data [Copperhill technologies 2020]

Byte number	Suspect Parameter	SPN
1	Engine Coolant Temperature	110
2	Fuel Temperature	174
3,4	Engine Oil Temperature	175
5,6	Turbocharger Oil Temperature	176
7	Engine Intercooler Temperature	52
8	Engine Intercooler Thermostat Opening	1134

Tabell 4.2: SPN-data [Copperhill technologies 2020]

J1939-meldingene kan i teorien avleses og tolkes manuelt ved hjelp av dokumentet SAE J1939/71, som inneholder oversikter over PGN-er og SPN-er sammen med informasjon om hvordan data skal transformeres. Dette blir raskt en omfattende jobb hvis større mengder data skal undersøkes, i praksis er det vanlig å benytte software til dette.

Dokumentet SAE 1939/13 definerer en standardkontakt for diagnostiseringsformål som er avbildet i figur 4.1. [Copperhill Technologies 2020] I J1939-kontakten (Deutsch HD10-9-1939) er pinne A og pinne B knyttet til systemets strømforsyning og pinne C og pinne D er henholdsvis CAN-high og CAN-low. Pinne E er tilknyttet deling av CAN-data. Pinne F og pinne G er knyttet til den spesifikke standarden som blir brukt for kommunikasjon mellom motorstyreenheter på tunge kjøretøy. De to siste pinnene, H og J, er mer frie for anvendelse for fabrikantene.



Figur 4.1: Deutsch HD10-9-1939-kontakt [Copperhill Technologies 2020]

4.3 OBD - On-board diagnostics

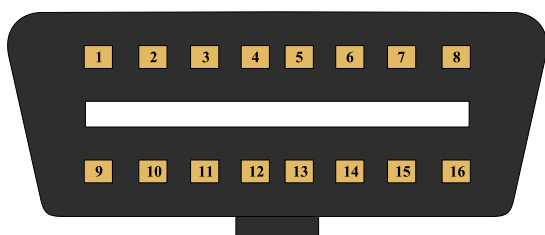
On-board diagnostics, OBD, er et diagnostiseringssystem som alle nyere kjøretøy innehar. Oppstår det uregelmessigheter ved et kjøretøy vil typisk en varselampe lyse, som indikerer at noe er galt. Med en OBDII-skanner kan OBD-porten tilkobles for å lokalisere og diagnostisere uregelmessigheten.

OBDII oppsto på midten av 90-tallet i USA og ble raskt anbefalt av Society of Automotive Engineers, SAE, som førte til en standardisering av diagnostiseringskoder og utforming av den fysiske porten på tvers av ulike kjøretøyfabrikanter. På tidlig 2000-tallet ble OBDII-standarden pålagt alle nye personkjøretøy i EU. Tunge kjøretøy i EU fikk det samme kravet i 2010. [CSS Electronics 2021b]

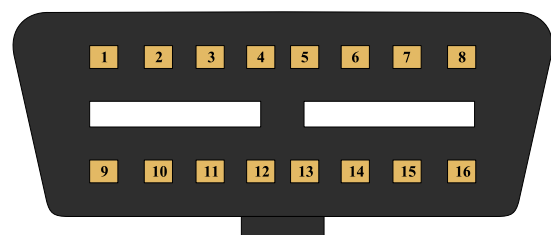
OBDII innehar et omfattende sett med standardiserte parameteridentifikasjoner som er praktisk sett fra et dataloggingsperspektiv. Dette gjør det mulig å få lesbar loggerdata fra OBD-porten for ulike parametere – uavhengig av kjøretøyfabrikant.

Spesifikasjonen for den standardiserte utformingen for OBDII er i form av en 16-pins J1962-kontakt, definert av SAE J1962. Pin nummer 6 og nummer 14 er henholdsvis CAN-high og CAN-low. Pin nummer 7 og 15 tilhører ISO 9141 og ISO 14230-4, hvor ISO 9141 er standarden som blir brukt for diagnostiseringssystemet og ISO 14230-4 tilhører utslippsrelaterte systemer. Strømforsyningen til systemet tilhører pin nummer 16 og er enten på 12 volt eller 24 volt. De resterende pin-ene er mer frie for anvendelse for fabrikantene, med unntak av pin 4 og 5 som brukes til jordingsformål. [Sawant og Mane 2018]

Lette kjøretøy anvender en spenning på 12 volt med utforming av J1962-kontakt som kan sees i figur 4.2, mens tunge kjøretøy anvender en spenning på 24 volt med utforming av J1962-kontakt som er illustrert i figur 4.3. Forskjellen i utforming mellom de to typene er den langsgående skillebiten mellom de to pin-radene som er hel for 12-voltssystemet og splittet for 24-voltssystemet. Skillebiten forhindrer feilkobling som kan skade kjøretøy og utstyr.



Figur 4.2: J1962-kontakt med 12-voltssystem (egenprodusert)



Figur 4.3: J1962-kontakt med 24-voltssystem (egenprodusert)

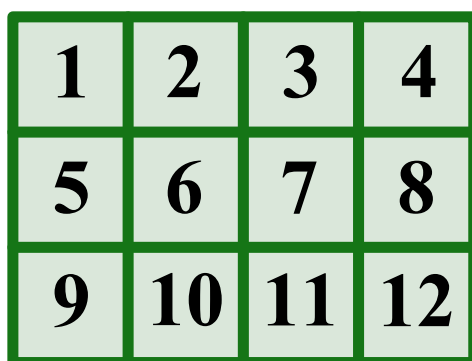
4.4 FMS - Fleet Management System

FMS er et standard flåtestyringsystem som ble først utviklet i 2002 ved et samarbeid mellom de seks lastebilfabrikantene Volvo, Scania, Iveco, MAN, DAF og Mercedes-Benz. Motivasjonen var å skape et system med et felles språk og grensesnitt på tvers av fabrikantene, slik at flåtestyring kunne være uavhengig av hvilke type kjøretøy som blir benyttet.

FMS har standard protokoller for applikasjonslaget (SAE J1939/71), datalinklaget (SAE J1939/21) og det fysiske laget (ISO 11898-2). Utformingen til den fysiske tilkoblingsenheten er imidlertid ikke standardisert, slik at selve porten for fysisk tilkobling kan variere fra fabrikant til fabrikant. [3Rtablet 2021]

Fra et dataloggingsperspektiv kan FMS kan være anvendbart ettersom det er flere parametre som er blitt gjort tilgjengelig for informasjonsuthenting fra fabrikantene. Eksempler på tilgjengelige parametre er drivstofforbruk, drivstofføkonomi, kjøretøyhastighet, turtall, girvalg og aksellast.

På tross av at FMS-utformingen er opp til hver enkelt fabrikant, definerer FMS-standarden sin egen fysiske utforming, som er en firkantet 12-pinnet grønn enhet. Denne er illustrert i figur 4.4. De fleste av pinnene er valgfrie for fabrikantene å benytte, men av standarden er pinne 12 strømforsyning, pinne 1 er jording og pinne 6 og 9 er henholdsvis CAN-high og CAN-low.



1	2	3	4
5	6	7	8
9	10	11	12

Figur 4.4: FMS-utforming fra standard (egenprodusert)

5 Loggerinstrumenter

Det eksisterer mange instrumenter for logging av kjøretøy, både for logging av posisjonsdata og motordata. I denne oppgaven er det blitt valgt å fokusere på produkter fra de to aktørene Racelogic og CSS Electronics for logging av henholdsvis posisjons- og motordata. Produkter fra nevnte aktører får gode tilbakemeldinger på prestasjon og brukervennlighet, samt at NTNU besitter relevante produkter fra disse aktørene som er aktuelle for forskningsarbeid. I kapittel 5 presenteres loggerinstrumenter og utstyr som har vært relevant for prosjektet.

5.1 Posisjonslogging med VBOX fra Racelogic

VBOX er en produktserie med dataloggere fra det britiske selskapet Racelogic. I produktserien finnes det en rekke varianter, hvor tilkoblingsmuligheter for tilleggsutstyr er det som i hovedsak skiller dem. I tillegg varierer dataloggerhastigheten mellom 5 Hz og 100 Hz for de ulike loggerne. [Racelogic 2020b] VBOX Mini, illustrert i figur 5.1, har mulighet for tilkobling av en kablet antenne for forbedret signal. Denne antennen monteres på taket av kjøretøy ved hjelp av en innebygd magnet. VBOX Mini starter logging ved registrert bevegelse, med en frekvens på 10 Hz. Posisjonsdata fra loggingen blir lagret direkte til et SD-minnekort.

Filformatet VBOX bruker er DBN, og kan leses i Racelogics eget program VBOX Test suite. Dataen i programmet kan fremstilles med grafer og posisjon i kart, og eksporteres til CSV-format for bruk i andre programmer som Excel og Python.

VBOX Mini har også et innebygget gyroskop som gir data for akselerasjon i horisontalplanet. Dette kan brukes for å estimere retningsendring i horisontalplanet og slip ratio. En mulighet er å benytte dette for å estimere friksjonsendring som en funksjon av horisontalkurvatur.



Figur 5.1: VBOX Mini [Racelogic 2020a]

5.2 Motordatalogging med CANedge1 fra CSS Electronics

CSS Electronics er et dansk firma som har spesialisert seg innenfor utvikling av helprofesjonelle, brukervennlige og kostnadseffektive CAN-bus motordataloggere og software, hvor loggerinstrumentet CANedge1 er illustrert i figur 5.2. Tekniske spesifikasjoner tilhørende CSS Electronics' CANedge1 er gitt i tabell 5.1.

Lengde [mm]	50,2
Bredde [mm]	83,4
Høyde [mm]	24,5
Vekt [g]	100
Driftstemperatur [°C]	-25 til 70
Statusindikasjoner	LED-lamper

Tabell 5.1: Tekniske spesifikasjoner tilhørende CSS Electronics' CANedge1 [CSS Electronics 2020a]



Figur 5.2: Loggerinstrumentet CANedge1 [CSS Electronics 2020b]



Figur 5.3: Adapterkabel av type 2, DB9 til Deutsch 9-pin [CSS Electronics 2020c]



Figur 5.4: Adapterkabel DB9 til OBDII [CSS Electronics 2021c]

Dataloggerinstrumentet CANedge1 har to fysiske CAN-bus grensesnitt, med standard DB9-tilknytning. For logging av tunge kjøretøy av nyere dato (i tråd med endringen i datahastighet kommentert i kapittel 4.2), brukes en adapterkabel av type 2 med DB9 til Deutsch HD10-9-1939-kontakt som er illustrert i figur 5.3, alternativt en standard OBDII-adapterkabel med DB9 til OBDII som er illustrert i figur 5.4.

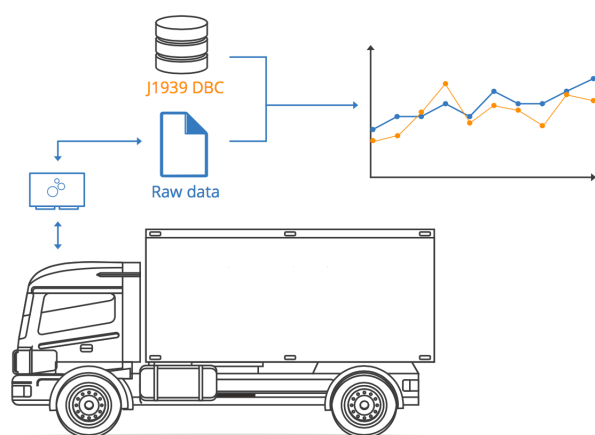
Et alternativ til adapterkablene er en CANCrocodile, en tilkoblingsløs CAN-dataleser som er illustrert i figur 5.5. [CSS Electronics 2021a] CANCrocodile krever ingen fysisk tilkobling til en noen port i kjøretøyet, og får enten strøm eksternt fra eksempelvis en sigarettenser, eller direkte fra CANedge1 som må være strømsatt. CANCrocodile klipses over ledningene CAN-high og CAN-low og registrerer rådata til CANedge1 ved bruk av elektromagnetiske felt.

En fordel ved å benytte CANCrocodile er fraværet av den fysiske tilkoblingen, noe som eliminerer all potensiell garantiprobmatikk knyttet opp mot den direkte tilkoblingen til kjøretøyets CAN-bus. En utfordring kan være å få tilgang til ledningene CAN-high og CAN-low som skal inn i klipsen. Ledningene har designerte pinner i Deutsch HD10-9-1939-kontakten, OBD-kontakten og den standardiserte FMS-utformingen, hvor det vil være naturlig å se etter tilgjengelighet.

CANedge1 støtter alle CAN-baserte protokoller, deriblant SAE J1939-standard. Logging og overføring av data baseres på kontrollmeldinger fra CAN-busen og blir satt opp i en standard MF4-fil, som kan avleses direkte på datamaskin. CANedge1 benytter et uttakbart SD-minnekort for oppsett av konfigurasjoner og loggerfiler. Frekvensen på dataloggingen kan utføres helt ned til hvert 50 μ s, med flere muligheter til å forhåndsdefinere ulike loggermønstre. CANedge1 har ikke støtte for posisjonsdata, men har en innebygget høypresisjons sanntidsklokke. Ved bruk av tidsstempler kan loggede motordata sammenstilles med posisjonsdata fra andre loggerinstrumenter som også har sanntidsklokke.



Figur 5.5: CANCrocodile med ekstern strømtilkobling fra CSS Electronics [CSS Electronics 2021a]



Figur 5.6: Datagang fra logging til fremstilling [CSS Electronics 2020d]

CSS Electronics har også utviklet software som automatiserer tolkning av rå CAN-data. De tilbyr en ferdigutviklet J1939 DBC-fil, som inneholder et sett med dekodingsregler som trengs for å konvertere loggerdata til menneskelig lesbar form. [CSS Electronics 2020d] Filen inneholder mer enn 1060 PGN-er og 6400 SPN-er, funksjonen til disse er presentert i kapittel 4.2. En illustrasjon av datagangen fra logging med CANedge1 til fremstilling av menneskelig lesbar data er illustrert i figur 5.6.

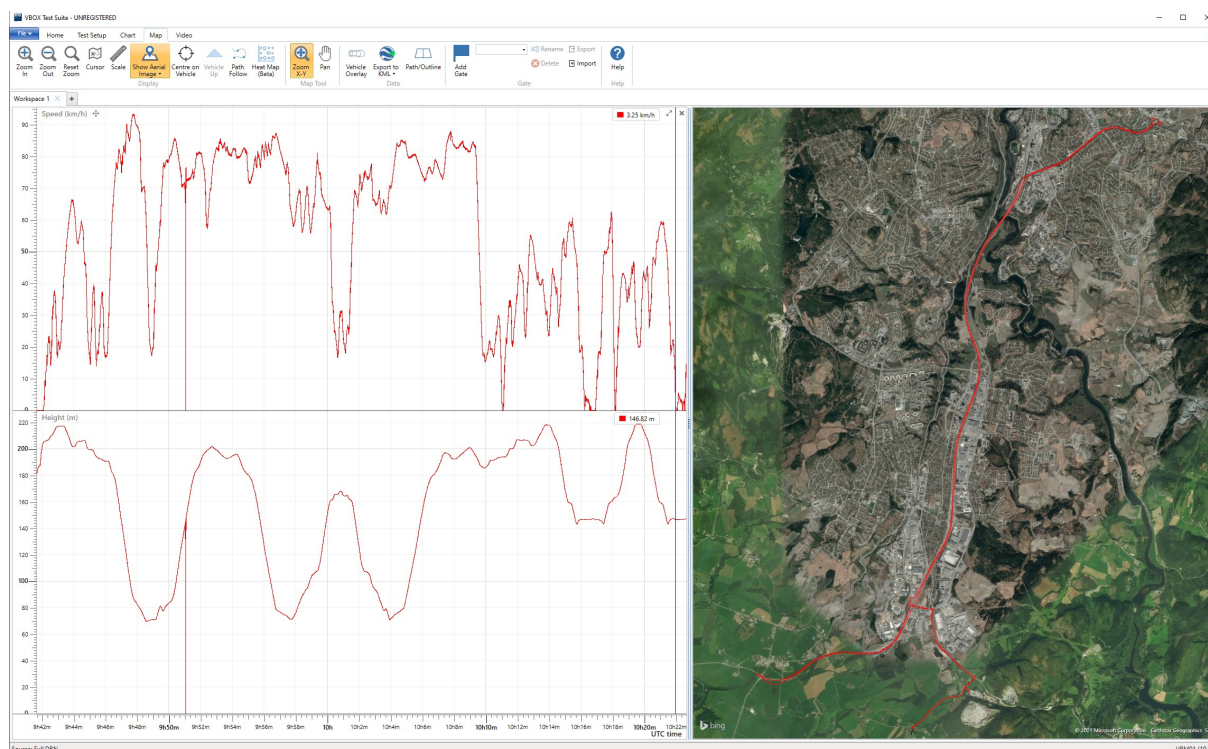
6 Fremstilling og sammenstilling av loggerdata

I kapittel 6 beskrives metodene for fremstilling og sammenstilling av logget posisjons- og motordata. To ulike metoder for fremstilling av høydeprofil presenteres. Til slutt introduseres hastighetsmodellen som tar inn både posisjons- og motordata for å generere hastighetsprofiler for tunge kjøretøy i motbakke. Mer detaljerte og dyptgående beskrivelser av hvordan hastighetsmodellen er bygget opp og videreutviklet fra utgangsmodellen til Børnes og Aakre (2011) er fremlagt i kapittel 10.

6.1 Fremstilling av posisjonsdata fra VBOX

Racelogic har egne programmer som kan presentere og analysere posisjonsdata fra VBOX. Programmene VBOX Test Suite og Circuit Tools kan presentere grafer for ulike loggekanaler og er åpne og gratis å nedlaste. VBOX Test Suite har en kartfunksjon som viser aktuell posisjon til en hver tid. Circuit Tools har muligheten for å eksportere hele loggerfiler til CSV-format. VBOX Test Suite er kun tilgjengelig på Windows, mens Circuit tools kan installeres på Windows, Mac OS og iOS for iPad og iPhone.

Posisjonsdata fra VBOX lagres på DBN-format, som er et rent tekstformat hvor innholdet struktureres ved bruk av punktum og parenteser. Programmene leser dataen, som fremstilles i form av grafer, statistikk og posisjonsdata i kart. Eksempel på fremstilling av hastighet-, høyde- og posisjonsdata i VBOX Test Suite er presentert i figur 6.1. Kart og grafer er dynamiske som gjør det enkelt å forstørre til ønsket oppløsning. Lokalisering ved bruk av kart i VBOX Test Suite og eksportering av data til CSV-format i Circuit Tools er interessante funksjoner for dette prosjektet.



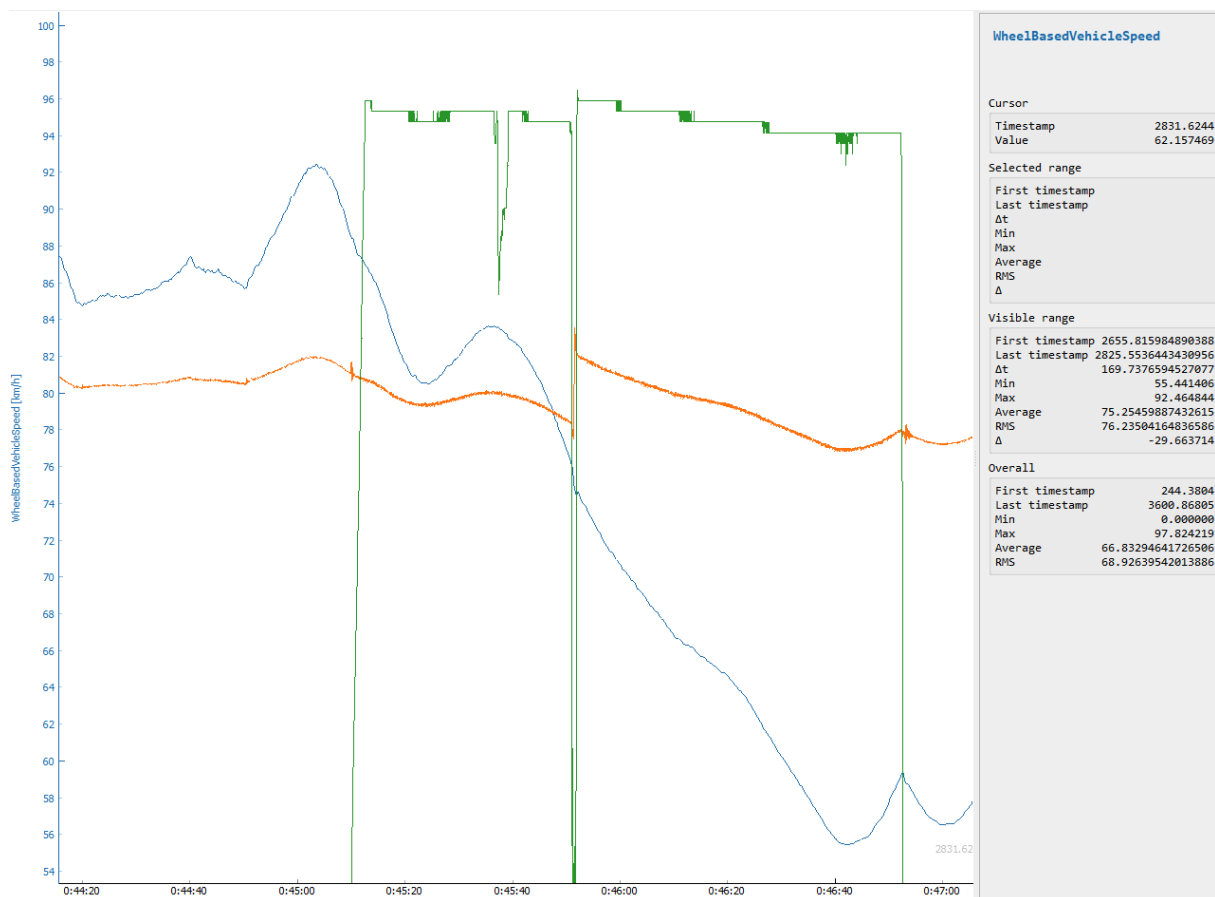
Figur 6.1: Eksempel på datafremstilling med VBOX Test Suite (Skjerm bilde fra VBOX Test Suite)

6.2 Fremstilling av motorloggerdata fra CANedge1

CSS Electronics anbefaler på sine sider programmet *asammdf GUI* som et verktøy for å konvertere DBC-filer, plote og eksportere data. Programmet er opprinnelig designet for ASAM (Association for Standardisation of Automation and Measuring Systems) for anvendelse på MDF-filer. (Measurement Data Format) GUI-biten (Graphical User Interface) går hovedsaklig på visualisering av data, sammen med databehandlingsfunksjoner som konvertering, eksportering, klipping, re-sampling og filtrering. [Hrisca, Daniel 2018] Programmet er kompatibelt med Windows, Mac OS og Linux, er åpent og gratis å nedlaste.

Rå CAN-data på heksadesimalt tallformat logget fra tunge kjøretøy med CANedge1 lagres i MF4-format. Loggerfilen åpnes i *asammdf GUI*, hvor en CAN-database må anvendes for å transformere dataen til menneskelig lesbar form. J1939-konverteringsfilen fra CSS Electronics som er beskrevet i kapittel 5.2 kan brukes til dette. Etter at en konverteringsfil er blitt anvendt på dataen, kommer en liste med gjenkjente PGN-er (kanaler) med kanalnavn. Kanalene kan fremstilles grafisk, enten isolert eller i kombinasjon med hverandre. I figur 6.2 er kanaler for dreiemoment (grønn graf), hastighet (blå graf) og turtall (oransje graf) sammenstilt i *asammdf GUI* over en liten del av en loggerfil. I figuren er y-aksen (km/t) kun korrekt for hastighet, derav samme farge, men dette er enkelt å bytte ved å klikke på de ulike kanalene. Alle kanalene er plottet mot tid på x-aksen.

Ved ønske om eksport av loggerdata for videre databehandling er det en egen fane for dette, hvor ønskede kanaler kan velges. Kanalene kan ved behov klippes til eller samples med en annen frekvens enn de originalt ble logget med. MDF og CSV er to av flere filformater data kan eksporteres til.



Figur 6.2: Eksempel på datafremstilling med asammdf GUI (Skjerm bilde fra asammdf GUI)

6.3 Fremstilling av høydeprofil

Det er blitt sett på to kilder til data for fremstilling av høydeprofil; data tilhørende senterlinjen for aktuelt vegstrekke i Nasjonal vegdatabank og loggerdata fra VBOX. Fullstendig kode for fremstilling av høydeprofil med data fra VBOX og NVDB finnes i henholdsvis vedlegg A og B. Vurdering og begrunnelse for valgt metode er beskrevet i kapittel 10.2.

6.3.1 Høydeprofil med data fra NVDB

En metode for å modellere høydeprofil er ved å bruke eksisterende høydedata fra NVDB, Nasjonal vegdatabank. x-, y- og z-koordinater hentes ut ved å benytte Vegkart, Statens Vegvesens kartløsning for NVDB-data. Koordinatene finnes ved å søke “stigning” i søkemotoren. For hver strekning er det gitt en objekt-id som inneholder beregningsgrunnlaget for strekningen. Koordinatene kopieres og lagres i en tekstfil. For å lese av koden i programspråket Python må alle verdier skilles med det samme skilletegnet. I rådataen er x-, y- og z-koordinatene skilt fra hverandre med komma og de ulike punktene er skilt fra hverandre med mellomrom. Det er ønskelig at alle verdier kun er skilt med komma, søk og erstatt-funksjonen er benyttet for å få til dette. Filen lagres på TXT-format.

Verdiene leses av et Python-skript, hvor første, andre og tredje verdi vil være x-, y- og z-

koordinatene til første punkt. Hver tredje verdi vil således gi x-, y og z-koordinat for neste målepunkt. De tre ulike koordinattypene lagres til egne lister. En for-løkke går gjennom listene og beregner horisontal avstand og høydeendring for hvert steg. De beregnede verdiene blir lagret til nye lister som kan plottes som høydeprofil og stigning.

6.3.2 Høydeprofil med data fra VBOX

En annen metode for å modellere høydeprofil er ved å benytte loggerdata fra VBOX. Den øverste linjen i CSV-filen beskriver innholdet til de respektive kolonnene under. Da stigninger er av interesse, er det hensiktsmessig å kun velge de radene som utgjør selve stigningen. Ved bruk av kart og tidsstempling i VBOX Test Suite, kan starten og slutten av stigninger i CSV-filen avgrenses. I Python-koden benyttes en funksjon som tar gjennomsnitt av ti og ti verdier for å jevne ut loggerstøy og redusere datamengden. For at dette skal fungere må den modifiserte CSV-filen alltid inneholde den første informasjonsraden og $n * 10 + 1$ rader, da det er forskjellen mellom to og to punkter som snittes.

I Python blir CSV-filen lest, og kolonnene med høydedata og data for horisontal avstand blir lagret i lister. Første rad beskriver innholdet i radene under. Kolonnen for høyde beskrives med 'Height' og avstand med 'Distance'. En for-løkke beregner stigningen mellom hver rad og lagrer verdien til en liste. Fra denne listen blir det laget en ny liste som inneholder gjennomsnittlig stigningsverdi for ti og ti punkter. Hvert tiende distansepunkt blir lagret i en egen liste og blir sammen med den gjennomsnittlige stigningslisten brukt som grunnlag for beregning.

6.4 Sammenstilling av data til hastighetsmodell

Posisjons- og hastighetsdata er sammenstilt til en hastighetsmodell i programspråket Python. For grafisk sammenstilling er det tatt utgangspunkt i visualiseringen Børnes og Aakre (2011) har brukt i Excel, eksemplifisert i figur 3.2. Hastighet og høyde opererer i samme diagram, men tilhører to ulike y-akser. Horisontal avstand går langs x-aksen, som er felles for å stedfeste hastighet og høyde.

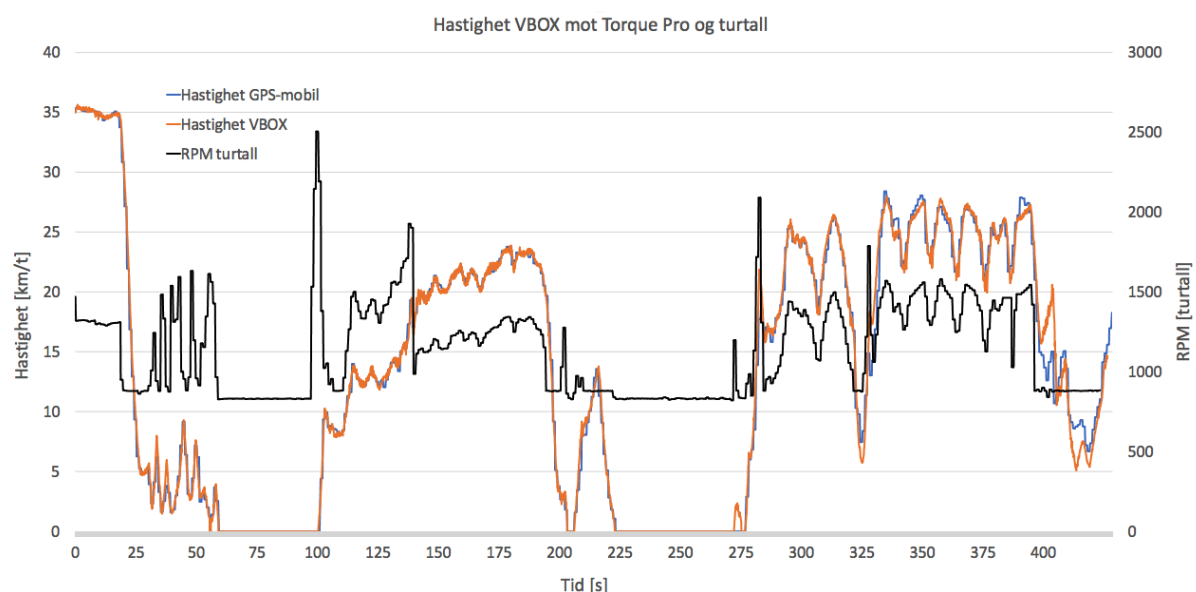
7 Forsøk med lette kjøretøy

Kapittel 7 omhandler utprøving av loggerutstyr på personkjøretøy. Posisjonslogging med VBOX fra Racelogic og motorlogging med CANedge1 fra CSS Electronics ble gjennomført i et tidlig stadie av prosjektet.

7.1 Utprøving av VBOX på personkjøretøy

Under forprosjektet til masteroppgaven ble det gitt tilgang på loggerutstyret VBOX fra Racelogic. Under pilotforsøk med personkjøretøy ble VBOX Mini brukt til å logge posisjonsdata, og appen Torque pro ble brukt til å logge motordata fra bilens OBD-port. Loggerfrekvensen ble satt til 10 ganger i sekundet for både VBOX og Torque pro. Dataen fra de ulike loggerinstrumentene ble eksportert til Excel i CSV-format, deretter sammenstilt ved hjelp av å sammenligne målt hastighet fra de to uavhengige loggerinstrumentene. Eksempel på uthentet data fra pilotforsøket kan betraktes i figur 7.1, hvor omdreininger

per sekund og hastighet er logget mot tid. Posisjonsdata fra pilotforsøket er plottet i kart fra Google Earth i figur 7.2.



Figur 7.1: Sammenstilt data fra VBOX og Torque pro fra pilotforsøk med personkjøretøy



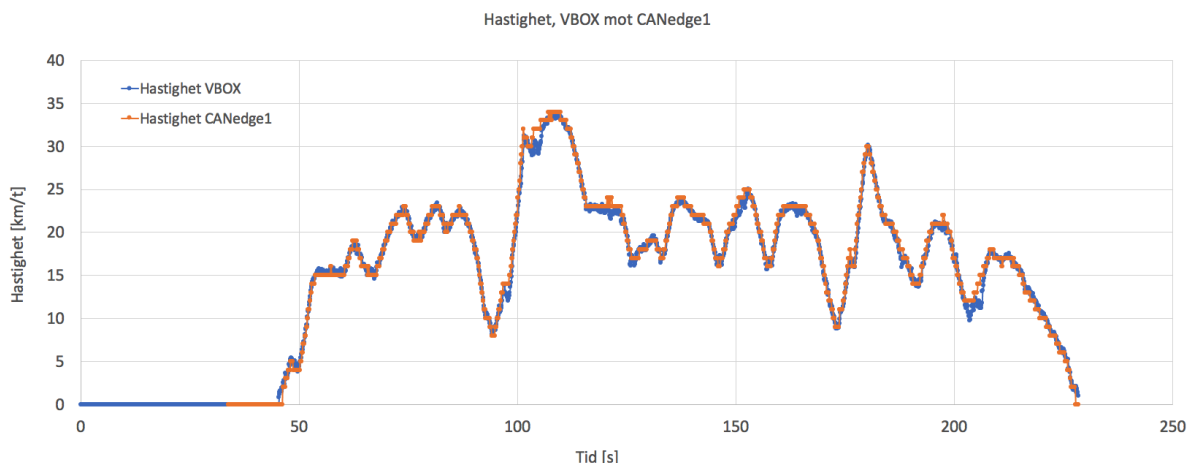
Figur 7.2: Kart med logget posisjonsdata fra første pilotforsøk med VBOX Mini (Skjerm-bilde fra Google Earth)

7.2 Utprøving av CANedge1 med VBOX på personkjøretøy

Det ble under forprosjektet til masteroppgaven kommunisert til veileder at motorloggerutstyret CANedge1 sammen med software levert av CSS Electronics kunne vært et praktisk verktøy for logging og fremstilling av motordata. Utstyret med software ble kjøpt inn og i første fase utprøvd på personkjøretøy i kombinasjon med VBOX Mini.

Det ble kjørt en runde før loggerdata fra de to respektive loggerinstrumentene ble uthentet og eksportert i CSV-format, for så å bli sammenstilt i Excel. Av tallene kom det frem at

tidsstempelen til dataen tilknyttet CANedge1 var forskjøvet sammenlignet med dataen fra VBOX. Kurvetilpasning av hastighetsprofilene fra de to loggerinstrumentene gav en tidsforskyvning på 4029 sekunder for CANedge1 sammenlignet med VBOX. Resultatet etter kurvetilpasningen kan sees i figur 7.3. Dette ble tatt hensyn til i konfigurasjonseditoren til CANedge1, hvor en manuell tidsforskyvning på sanntidsklokken kan legges inn på konfigurasjonsfilen på minnekortet hvor data lagres til. Hva som er grunnen til forskjellen i tidsstempelen mellom loggerinstrumentene er usikkert. Fra prosjektets dataloggingsperspektiv er det mindre kritisk at tidsstempelen er korrekt, så lenge tidsstemplene fra de to loggerinstrumentene samsvarer.



Figur 7.3: Kurvetilpasning av hastighetsprofiler fra loggerforsøk med personbil, CANedge1 og VBOX

Standard konfigurasjonsfil på minnekortet til CANedge1 er tilknyttet J1939-standarden og er tilpasset logging av motordata fra tunge kjøretøy. Denne filen ble derfor under forsøket erstattet med en annen konfigurasjonsfil tilgjengelig på CSS Electronics hjemmesider, tilpasset logging av motordata fra personkjøretøy.

7.3 Hensikt med pilotforsøk på personkjøretøy

Hensikten med å gjennomføre pilotforsøk på lette kjøretøy var å bli komfortabel med utstyret som skulle tas i bruk ved logging av data fra tunge kjøretøy. Det var viktig for prosjektgruppa å få avdekket potensielle tekniske utfordringer med loggerutstyret så tidlig som mulig, slik at dette kunne bli tatt hånd om. Det var uvisst hvor mange anledninger det ble til å logge data fra tunge kjøretøy, derfor var det viktig å være godt forberedt under forsøkene som kom senere i prosjektet. Pilotforsøkene gav i tillegg en indikasjon på hva som ventet av databehandling.

8 Forsøk med tunge kjøretøy

Forsøk som ble gjennomført med tunge kjøretøy legges frem i kapittel 8. Det ble først gjort loggerforsøk på lokale lastebiler ved NTNU, før henholdsvis transportfirma og forhandler bidro til å kunne gjennomføre lengre loggerseanser. Totalt ble det utført loggerforsøk på lastebilmodeller fra de tre fabrikantene Scania, Mercedes-Benz og Volvo. De ulike loggerforsøkene er beskrevet hver for seg, med bakgrunn, tilnærming og resultat. Informasjon og tekniske beskrivelser tilhørende de ulike lastebilene følger også i dette kapitlet.

8.1 Forsøk med lastebil ved NTNU

8.1.1 Bakgrunn

Etter gjennomførte forsøk på personkjøretøy, var det neste steget å gjøre forsøk på tunge kjøretøy. Resultatene fra forsøkene med personkjøretøy var relativt forutsigbare, i tillegg til at motorloggerutstyret fra CSS Electronics fungerte slik som beskrevet. Det var likevel ingen garanti for at overgangen til motordatalogging fra tunge kjøretøy skulle bli sømløs. Prosjektgruppa anså det derfor som viktig å gjennomføre motorloggerforsøk på tunge kjøretøy i et tidlig stadie i oppgaven, ettersom dette har vist seg å være utfordrende i tidligere studier.

Det ble tidlig etablert kontakt med transportgruppa ved NTNU, som ble orientert om masteroppgaven. Transportgruppa ved NTNU besitter flere lastebiler som brukes til drift- og vedlikeholdsformål ved universitetet. Prosjektgruppa mente dette ville være et naturlig sted å forhøre seg om muligheten for å gjennomføre motorloggerforsøk, ettersom transportgruppa er lokalisert nært universitetet, besitter flere typer kjøretøy, samt tidligere har bistått ved forskningsarbeid. Transportgruppa responderte positivt på henvendelsen og stilte lastebilene sine til disposisjon for motorloggerforsøk etter avtale.

8.1.2 Volvo FM 450, 2016-modell

Det første motorloggerforsøket ble utført på en søppelbil, en Volvo FM 450, 2016-modell som er illustrert i figur 8.1. I utgangspunktet var intensjonen å lokalisere J1939-kontakten og benytte adapterkabelen av type 2 for datatilkobling. Frontdeksler på fører- og passasjerside i førerhytta ble demontert, men noen J1939-kontakt ble ikke funnet. OBD-porten var imidlertid tilgjengelig, og kan sees øverst i figur 8.2. Illustrasjonen er fra demonteringen av frontdekselet på førersiden.



Figur 8.1: Volvo FM 450, 2016-modell (Foto: Nygaard og Kogstad)

CANedge1 ble tilkoblet OBD-porten med en OBDII-kontakt, hvor det grønne lyset som indikerer strømtilførsel straks ble opplyst. Lyset som indikerer CAN-tilkobling forble imidlertid slukket. Etter en liten kjøretur med loggerutstyret tilkoblet OBD-porten ble minnekortet tatt ut for å se om noe data hadde blitt registrert. Ingen loggerfiler ble lagret, CANedge1 hadde ikke klart å registrere noe data.

Det kan være flere grunner til at CANedge1 ikke klarte å lese data fra OBD-porten til Volvoen. En mulighet er at loggerutstyret var feilinnstilt, men dette er lite trolig da CANedge1 ikke så ut til å motta noen CAN-signaler i det hele tatt. En annen teori er at Volvo har deaktivert pinene som en standardinnstilling hvor de kun kan åpnes med en nøkkel som typisk diagnostiseringsverktøy innehar. Prosjektgruppa var i etterkant i kontakt med et merkeverksted som bekreftet at de benyttet OBD-porten til diagnostisering på denne typen kjøretøy, samt Volvo sentralt som uttalte at de hadde mer tro på en kobling opp mot flåtestyringssystemet for registrering av loggerdata.



Figur 8.2: Bilde av Volvo FM 450, 2016-modell etter demontering av deksel på førerside (Foto: Nygaard og Kogstad)

8.1.3 Mercedes-Benz 2536L/45, 2012-modell

Prosjektgruppa fikk også mulighet av transportgruppa ved NTNU til å teste ut motor-loggerutstyret på en containerbil, en Mercedes-Benz 2536L/45, 2012-modell, som kan sees i figur 8.3.



Figur 8.3: Mercedes-Benz 2536L/45, 2012-modell (Foto: Nygaard og Kogstad)

På denne lastebilen hadde prosjektgruppa ikke anledning til å fjerne frontdeksler for å lokalisere J1939-kontakten, men OBD-porten ble lokalisert på passasjersiden og tilkoblet med CANedge1. I dette tilfellet blinket lyset som indikerte tilkobling til CAN-busen i motsetning til forsøket med Volvo FM 450 hvor denne lampen var slukket. Etter en kort forsøksstur ble minnekortet uthentet, som viste en lagret loggerfil på omlag 14 megabyte, med over 80 PGN-er. Størrelsen på loggerfilen indikerte at langt mer data hadde blitt logget i forhold til forsøksturene på personbil, som typisk var på noen hundre kilobyte, med i underkant av 20 PGN-er.

J1939-konverteringsfilen fra CSS Electronics ble anvendt for å transformere loggerdataen fra heksadesimalt tallformat til lesbar form. Det viste seg at konverteringsfilen ikke klarte å gjenkjenne en eneste PGN for transformasjon. Ansvarlig i CSS Electronics uttalte i etterkant av å ha mottatt og inspisert loggerfilen at de aldri hadde sett et lignende scenario hvor en slik mengde PGN-er har blitt logget, hvorpå ingen data har blitt gjenkjent og transformert. Alderen på kjøretøyet ble presentert som den potensielle hovedgrunnen, da det som kjent ble gjort endringer i standarden for datakommunikasjonen i tunge kjøretøy i 2015-2016.

8.2 Forsøk med Scania fra Børstad Transport

8.2.1 Bakgrunn

Tidlig i mars tok prosjektgruppa kontakt med Børstad Transport AS, som har kontor på Sutterøya i Stjørdal. Børstad har tidligere bistått ved å tilgjengeliggjøre kjøretøy for arbeid med masteroppgaver ved NTNU og responderte positivt på henvendelsen om muligheten for å gjøre det samme for dette prosjektet.

I etterkant av forsøkene med lastebil ved NTNU som ikke gav ønsket motorloggerdata, var neste mål å forsøke å anvende CANCrocodile i forbindelse med et FMS-system i en lastebil. NTNU benytter ikke FMS-system på sine kjøretøy, derfor ble et transportfirma som benytter dette involvert for å få til et slikt forsøk.

Prosjektgruppa drøftet med Børstad hypotesen om større sannsynlighet for vellykket motordatalogging i forbindelse med FMS-systemet, en oppfatning Børstad delte. Datagangen inn i FMS-systemet er mindre avhengig av hver enkelt kjøretøyfabrikant, i motsetning til eksempelvis OBD-porten som fabrikantene kan prege i større grad.

Under besøket hos Børstad kunne en Scania-ansatt informere om at de tidligere hadde gjennomført vellykkede motorloggerseanser via OBD-porten. Dette tydet på at porten kunne være mer aktuell for motordatalogging i Scania enn forespeilet etter forsøkene ved NTNU med Volvo og Mercedes-Benz.

8.2.2 Scania R580, 2019-modell

Børstad Transport gav prosjektgruppa anledning til å utføre loggerforsøk på en Scania R580 2019-modell, hvor nøkkelparametere er gitt i tabell 8.1. Vogntoget er illustrert i figur 8.4 og benyttes daglig til å kjøre kraftfôr til gårder i Trøndelag for Felleskjøpet. Totalt ble det utført to forsøksturer på to ulike dager i midten av mars. Under begge forsøksturene var vogntoget fullastet og fulgte normal kjøreplan, med en totalvekt på 50,6 tonn.

Motorloggerutstyret CANedge1 ble under begge etappene tilkoblet i OBD-porten i kjøretøyet. Lysindikatoren for strømtilførsel og blinkingen som indikerer CAN-tilkobling startet umiddelbart, som tydet på at CANedge1 fikk tilgang på data. VBOX Mini ble koblet opp med den magnetiske antennen festet på taket av førerhytta. Strekningen Stjørdal - Trondheim ble kjørt på Europaveg 6 under det første forsøket, mens strekningen Åsen - Klett på samme veg ble kjørt under det andre forsøket. Begge forsøkene ble gjennomført på bar veg midt på dagen, typisk to timer før ettermiddagsrushet normalt inntreffer. Forsøkene ble filmet fra en personbil som fulgte tett bak vogntoget for å dokumentere eventuelle forhold som har påvirkning på hastigheten, eksempelvis trafikk og vegføre.

Modell	Scania R580
Årsmode	2019
Registreringsnummer	VJ17171
Egenvekt [kg]	14 880
Totalvekt [kg]	50 600
Motoreffekt [kW/HK]	427/581
Topphastighet [km/t]	90

Tabell 8.1: Nøkkelparametere tilhørende Scania R580, 2019-modell



Figur 8.4: Scania R580, 2019-modell med henger (Foto: Nygaard og Kogstad)

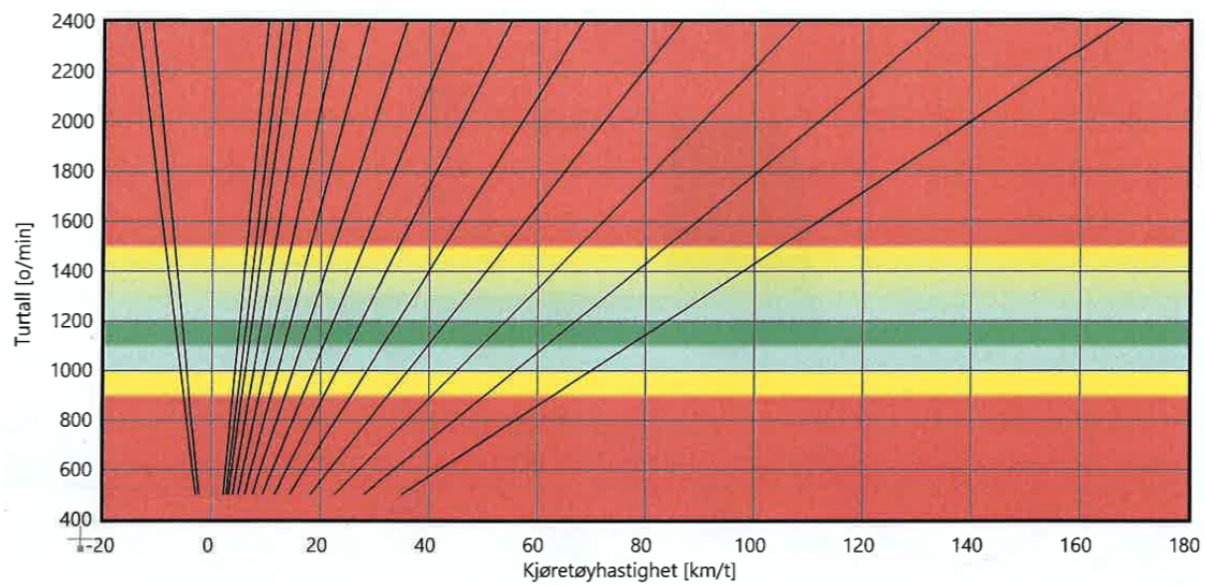
8.2.3 Resultater

Forsøksturen fra Stjørdal til Trondheim varte i totalt 40 minutter og gav en motorloggerfil på omlag 36 megabyte, mens forsøksturen fra Åsen til Klett varte i totalt 70 minutter og gav en motorloggerfil på omlag 62 megabyte. Dataen ble transformert med J1939-konverteringsfilen fra CSS Electronics, hvor 92 PGN-er ble gjenkjent. Et utvalg av relevante gjenkjente PGN-er (kanaler) med tilhørende beskrivelse som er aktuelle for implementering for optimalisering og utvidelse av eksisterende hastighetsmodell er listet i tabell 8.2.

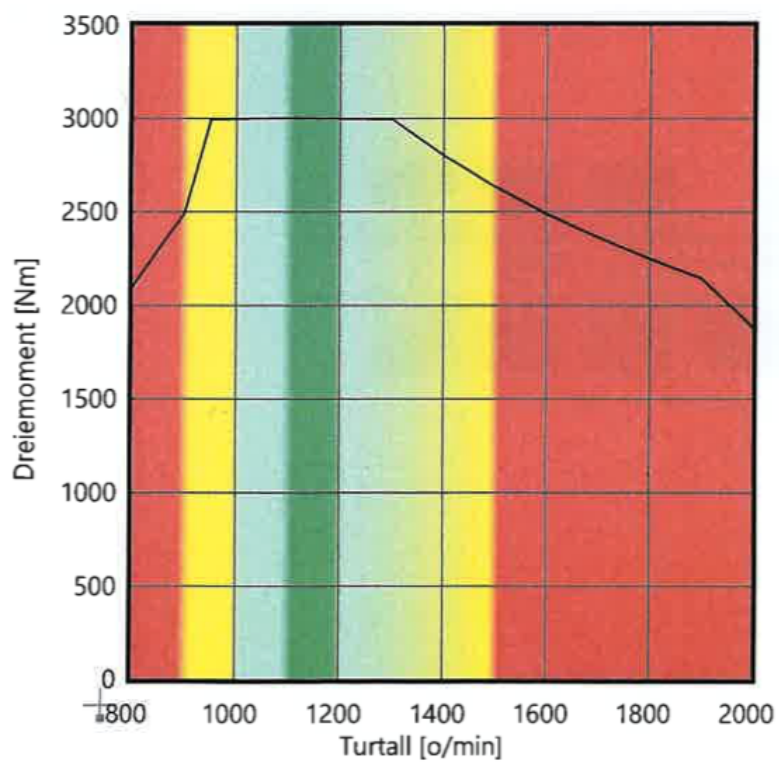
Kanalnavn	Beskrivelse
ActualEnginePercentTorque	Motorens kalkulerede dreiemoment gitt i prosent av dreiemomentets referanseverdi ved gitt turtall
EngineSpeed	Motorens omdreiningshastighet kalkulert over en minimum veivakselvinkel på 720 grader delt på antall sylindere
WheelBasedVehicleSpeed	Kjøretøyets hastighet kalkulert fra dekk eller bakaksel
EngineFuelRate	Drivstoffmengde brukt per tidsenhet
Latitude	Kjøretøyets breddegradsposisjon. Negative verdier er sør, positive verdier er nord
Longitude	Kjøretøyets lengdegradsposisjon. Negative verdier er vest, positive verdier er øst

Tabell 8.2: Utvalg av relevante gjenkjente PGN-er etter loggerforsøk med Scania

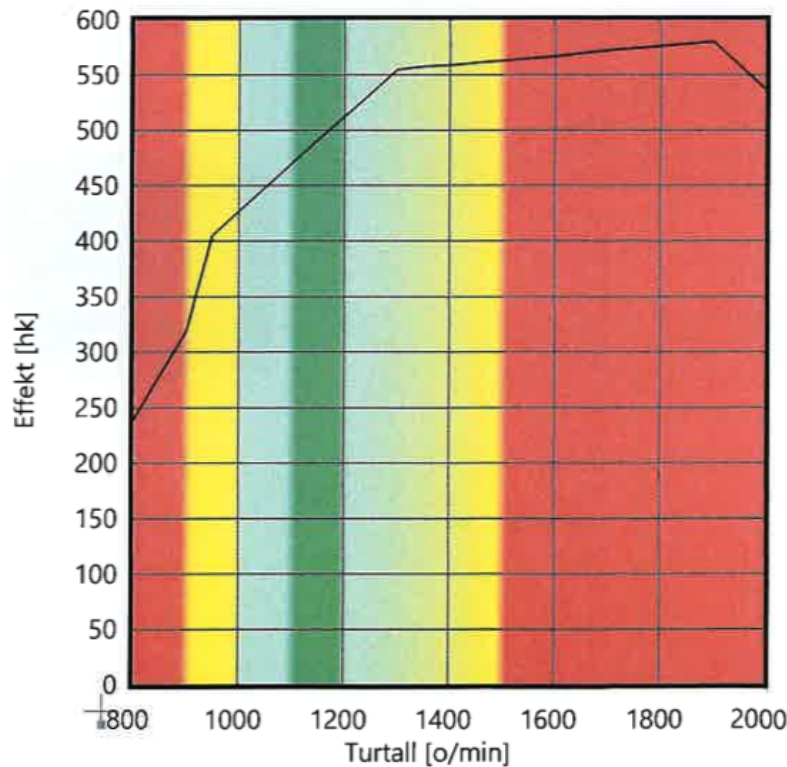
I etterkant av forsøksturene fikk prosjektgruppa tilsendt utvekslingskurver for girskift, dreiemoment- og effektkurve for lastebilen som ble brukt. Gjeldende gir er ikke blant tilgjengelige PGN-er for logging, så utvekslingskurvene for girskift er et nyttig verktøy for å implementere en girfaktor i hastighetsmodellen. Kurvene for dreiemoment og effekt gir en sammenheng mellom disse parameterene og motorens turtall. Scania R580s utvekslingskurve for girskift er vedlagt i figur 8.5, hvor de svarte strekene er gir, to for revers, to krabbegir og 12 ordinære. Det grønne feltet er motorens foretrukne turtall. Kurvene for dreiemoment og effekt er vedlagt i figur 8.6 og 8.7.



Figur 8.5: Utvekslingskurver for girskift, Scania R580, 2019-modell [Scania 2019]



Figur 8.6: Dreiemomentkurve, Scania R580, 2019-modell [Scania 2019]



Figur 8.7: Effektkurve, Scania R 580, 2019-modell [Scania 2019]

8.3 Forsøk med Mercedes-Benz i regi av NTNU

8.3.1 Bakgrunn

I etterkant av forsøkene med Scania fra Børstad, var det ønske i prosjektgruppa om å forsøke logging på et kjøretøy fra en annen lastebilfabrikant. Dette for å ha muligheten til gjøre hastighetsmodellen mer generell på tvers av ulike fabrikanter, samt få kartlagt likheter og ulikheter. I den forbindelse ble det også drøftet muligheten for å variere nyttelast over en fastsatt forsøksrute, for å se effekten ulik vekt har på oppnådd hastighetsprofil. Instituttet foreslo en løsning hvor en lastebil ble leid inn over en dag, med påfølgende større mulighet til å utforme egne loggerforsøk sammenlignet med det som hadde blitt gjort hos transportfirma. Det ble avtalt å gjennomføre leie av en Mercedes-Benz Arocs fra Bertel O. Steen i midten av April.

8.3.2 Mercedes-Benz Arocs, 2021-modell

Loggerforsøkene foregikk med en nyregistrert Mercedes-Benz Arocs tippbil, 2021-modell, som er avbildet i figur 8.8. Nøkkelparametere tilhørende lastebilen er gitt i tabell 8.3. Tidligere loggerforsøk med Mercedes-Benz ved NTNU som er beskrevet i kapittel 8.1.3, viste at det kan være utfordrende å få logget og uthentet motordata. Prosjektgruppa bestemte derfor å besøke Bertel O. Steen i forkant av dagen hvor lastebilen skulle leies, for å kartlegge loggermulighetene på forhånd.



Figur 8.8: Mercedes-Benz Arocs, 2021-modell (Foto: Nygaard og Kogstad)

Modell	Mercedes-Benz Arocs
Årsmodell	2021
Registreringsnummer	VJ24274
Egenvekt [kg]	13 700
Totalvekt [kg]	20 950, 28 000
Motoreffekt [kW/HK]	425/578
Topp hastighet [km/t]	90

Tabell 8.3: Nøkkelparametere tilhørende Mercedes-Benz Arocs, 2021-modell

I motsetning til Scania, fungerte ikke OBD-porten til motordatalogging på Mercedes-Benz' lastebil. Lampen som indikerer CAN-tilkobling på CANedge1 forble slukket, som betyr at ingen data var tilgjengelig for logging. Muligheten å filme dashbordet manuelt under kjøring i kombinasjon med logging fra VBOX ble foreslått, før forutsetningene for at CANCrocodile skulle fungere ble presentert for en tekniker. CANCrocodile er beskrevet i kapittel 5.2. Teknikeren lokaliserte CAN-high og CAN-low ledningene, i forbindelse med FMS-systemet i lastebilen. CANCrocodile ble klipset over de to ledningene som er illustrert i figur 8.9 og videre tilkoblet CANedge1. Lampen som indikerer CAN-tilkobling lyste i dette tilfellet og en liten loggerfil ble lagret og inspisert. Filen bekreftet at motordata hadde blitt logget og lagret.



Figur 8.9: CANCrocodyl montert i forbindelse med FMS-system (Foto: Nygaard og Kogstad)

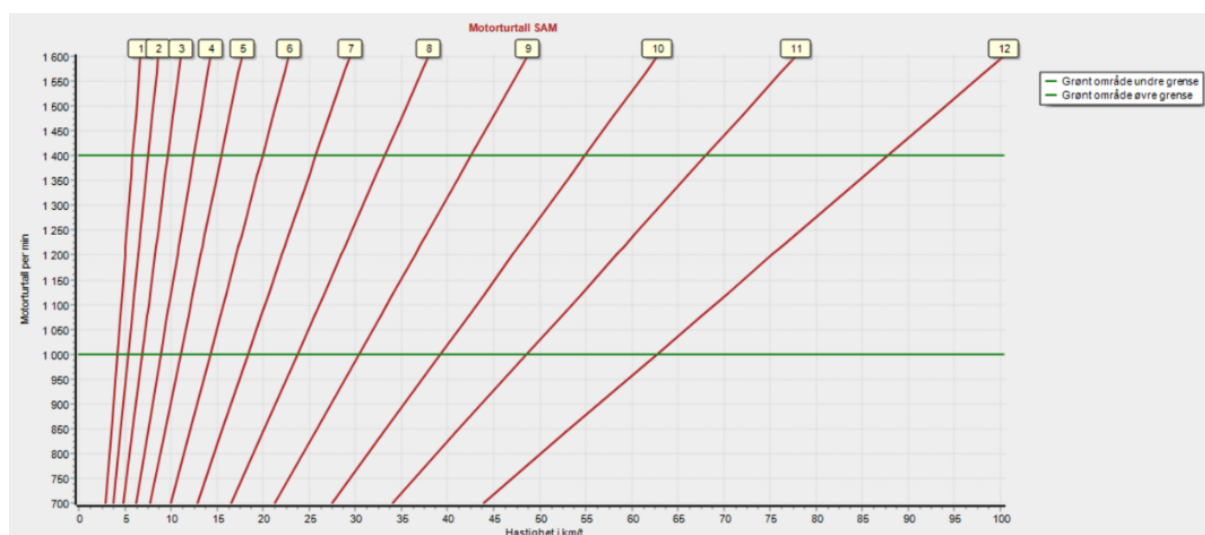
Under loggerforsøkene ble det kjørt med to ulike vektkombinasjoner, halvfull og full, med henholdsvis 20 950 kg og 28 000 kg totalvekt. Lasten var i form av grus som ble lånt ved Franzefoss Pukk, avdeling Vassfjell. VBOX Mini ble satt opp for å logge posisjonsdata på samme måte som under forsøkene med Scania, mens CANedge1 logget motordata ved hjelp av CANCrocodyl. Ruten som ble kjørt gikk fra pukkverket til Klett, før Europaveg 6 ble kjørt fra Klett til toppen av Moholtbakken. Derfra ble det snudd og kjørt tilbake på Europaveg 6 og til pukkverket igjen. Bakgrunnen for rutevalget er ønsket om kjøring i lange motbakker hvor terminal krabbehastighet blir nådd. Både Klettbakken, Okstadbakken og Moholtbakken er lengre motbakker hvor det er gode sjanser for at krabbehastighet nås.

Under forsøksturene satt det på en passasjer i lastebilen i tillegg til sjåføren som i hovedsak noterte føre- og trafikkforhold. Forsøksturene ble gjennomført på formiddagen under oppholdsvær og lastebilen fikk kjøre motbakkene fritt uten nevneverdig påvirkning av trafikken rundt.

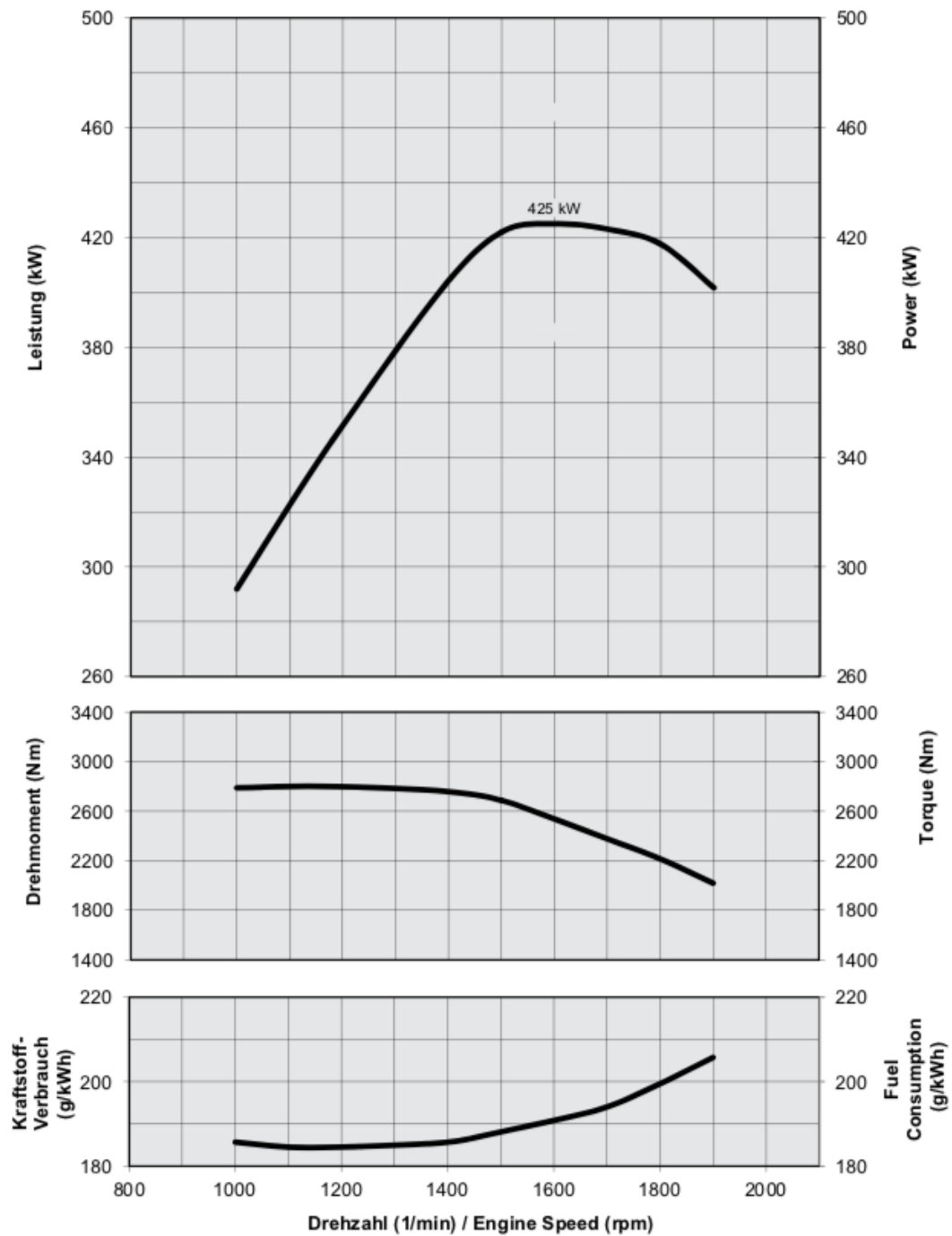
8.3.3 Resultater

De ble gjennomført to forsøksturer over beskrevet rute, med vekt på 20 950 kg og 28 000 kg. Forsøkene varte omlag 40 minutter, med tilhørende motorloggerfiler på i underkant av 30 megabyte. 75 PGN-er ble gjenkjent og transformert med J1939-konverteringsfilen fra CSS Electronics, som er noe færre kanaler sammenlignet med Scantias 92. Relevante felles kanaler for begge fabrikantene er *ActualEnginePercentTorque*, *EngineSpeed*, *Latitude* og *Longitude*. Hvor Scania beregner hastighet fra bakakselen, har Mercedes-Benz kanalen *FrontAxleSpeed*, hvor hastigheten beregnes ved et gjennomsnitt fra de to dekkene på fremakselen.

I etterkant av loggerforsøkene med Mercedes-Benz Arocs, tok prosjektgruppa kontakt med Mercedes-Benz sentralt for å få tilgang på motorspesifikke data. Girutvekslingskurven for den spesifikke modellen ble oversendt og er vedlagt i figur 8.10. Her er kun de 12 girene som gir fremdrift i rett retning illustrert. Effekt- og dreiemomentkurve for motoren ligger åpent på nettsidene til Mercedes-Benz og er vedlagt i figur 8.11. Som beskrevet for Scania, er grafene nødvendige for å implementere en realistisk girfaktor i en utvidet hastighetsmodell.



Figur 8.10: Girutvekslingskurve, Mercedes-Benz Arocs, 2021-modell [Mercedes-Benz 2021]



Figur 8.11: Effekt- og dreiemomentkurve, Mercedes-Benz Arocs, 2021-modell [Mercedes-Benz 2021]

8.4 Forsøk med Volvo fra Børstad Transport

8.4.1 Bakgrunn

Prosjektgruppa fikk i slutten av April etter samtaler med Børstad Transport anledning til å utføre loggerforsøk på et kjøretøy fra en tredje lastebilfabrikant, en Volvo. Etter forsøkene med Scania og Mercedes-Benz var det et ønske om å få til loggerforsøk med en siste lastebilfabrikant, for å kunne gjøre hastighetsmodellen ytterligere generell på tvers av ulike fabrikanter.

8.4.2 Volvo FH 540, 2019-modell

Loggerforsøket skulle utføres på en Volvo FH 540 2019-modell, illustrert i figur 8.12, hvor nøkkelparametere er gitt i tabell 8.4. I likhet med forsøket som ble gjennomført med Scania, blir vogntoget brukt til transport av kraftfôr til gårder i Trøndelag for Felleskjøpet.

Ved oppstart ble VBOX Mini plassert på taket av førerhytta for posisjonslogging. CANedge1 ble tilkoblet OBD-porten, men fikk ingen CAN-kontakt for motordatalogging. Det var under forsøket ikke anledning til å lete frem CAN-high og CAN-low ledningene i forbindelse med FMS-systemet, slik som det ble gjort under forsøket med Mercedes-Benz. Løsningen ble å filme dashbordet for å føre opp data knyttet til turtall, kraftpådrag og gir i etterkant.

Fraværet av elektronisk motordatalogging, samt vogntogets kjøreplan, gjorde at loggerforsøket ble noe kortere enn tiltenkt. Vogntoget skulle sørover fra Trondheim på Europaveg 6, hvor det er langt mellom lengre motbakker i forhold til i nordgående retning. Strekningen Trondheim - Klett ble filmet og logget med VBOX, med fokus på Okstadbakken. Forsøket ble gjennomført på bar veg tidlig om morgenen før morgenrushet setter inn, og vogntoget fikk kjøre uhindret av trafikk opp Okstadbakken.



Figur 8.12: Volvo FH 540, 2019-modell med henger (Foto: Nygaard og Kogstad)

Modell	Volvo FH 540
Årsmodell	2019
Registreringsnummer	VJ11540
Egenvekt [kg]	12 250
Totalvekt [kg]	50 000
Motoreffekt [kW/HK]	397/540
Topp hastighet [km/t]	90

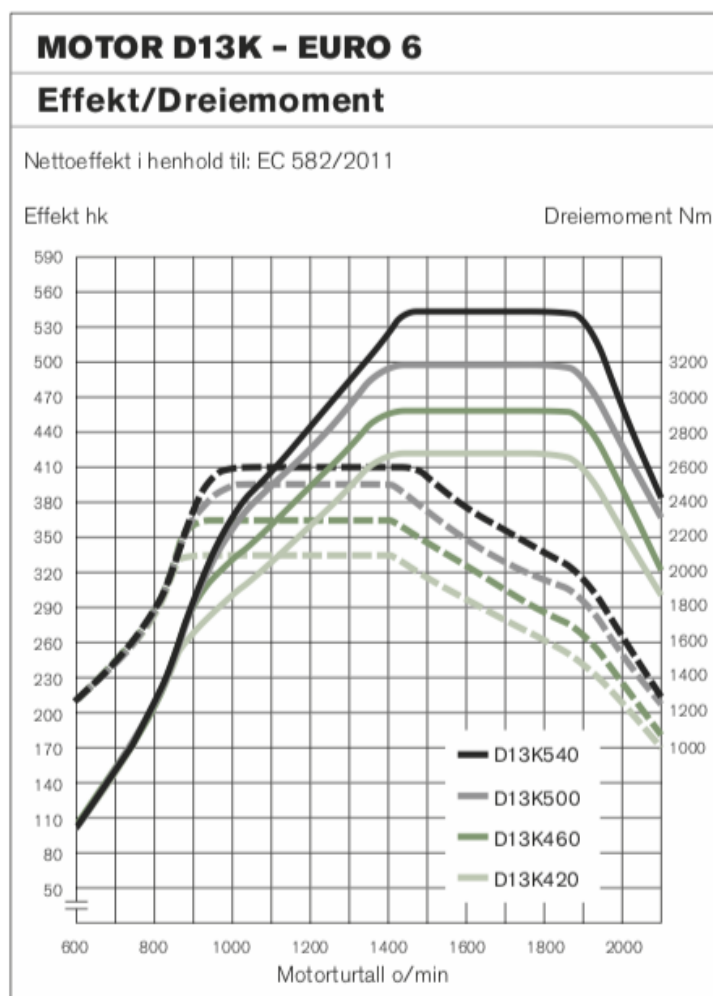
Tabell 8.4: Nøkkelparametere tilhørende Volvo FH 540, 2019-modell

8.4.3 Resultater

Hele loggerforsøket varte i overkant av 20 minutter, hvor dashbordet ble filmet i 3 minutter fra bunn til topp av Okstadbakken. Dashbordet gir informasjon om turtall, gir og hastighet, som er viktige parametere for utvikling av hastighetsmodellen. Sjøfører informerte underveis i bakken om maksimalt kraftpådrag, som sammen med gjeldende turtall bør reflekteres i teoretiske effekt- og dreiemomentkurver for aktuell motor.

Effekt- og dreiemomentkurver for Volvos D13K-motorer ligger åpent på Volvos hjemmesider og er illustrert i figur 8.13. Volvo FM benytter motor av typen D13K, og modellen med 397 kW som ble brukt i forsøket tilhører de svarte kurvene i diagrammet.

Prosjektgruppa tok kontakt med Volvo i etterkant av forsøket som oversendte utvekslingstall for hvert av de 12 girene i den aktuelle motoren. Det ble gitt på tallformat, så girutvekslingskurvene tilsvarende for Scania og Mercedes-Benz kan fremstilles ved behov.



Figur 8.13: Effekt- og dreiemomentkurver, D13K-motor fra Volvo [Volvo 2021]

9 Sammenfatning av forsøk med tunge kjøretøy

Det ble gjennomført loggerforsøk på totalt 3 ulike lastebilfabrikanter. Med Scania ble det gjennomført to lengre forsøksturer på Europaveg 6, med fullastet vogntog under normal kjøreplan for Børstad Transport. En Mercedes-Benz ble leid over en dag, hvor en fast sløyfe på Europaveg 6 ble kjørt med ulike vektkombinasjoner. Det ble en kort forsøkstur med Volvo fra Børstad Transport, i likhet med Scania et fullastet vogntog under normal kjøreplan.

9.1 Datainnsamling

For posisjonslogging ble VBOX Mini brukt under alle forsøksturene. Den magnetiske antennen ble festet på taket av førerhytta, ledningen klemte i passasjerdøren, mens en powerbank ble benyttet for å gi strøm til utstyret som ble lagt på passasjersiden. VBOX fungerte som forventet, med forutsigbare loggerfiler etter hvert forsøk.

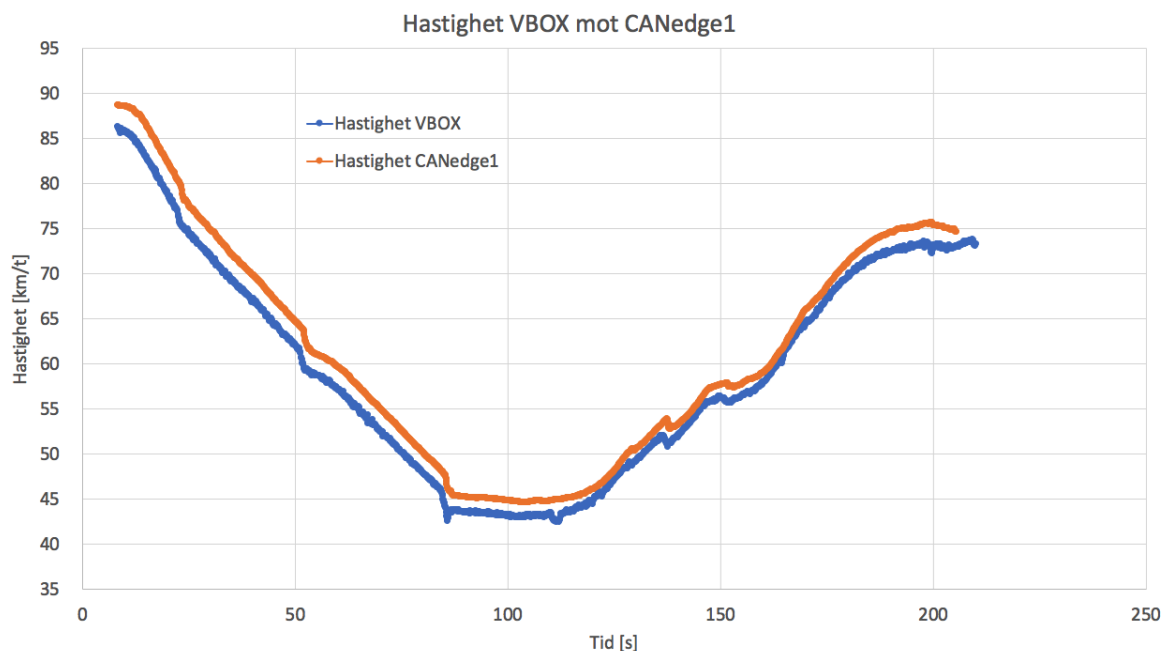
Tre ulike metoder ble brukt for motordatalogging. For Scania var OBD-porten tilgjengelig for CAN-tilkobling, så denne ble brukt for logging med CANedge1. For Mercedes-Benz var OBD-porten ikke tilgjengelig for CAN-tilkobling. Ledningene CAN-high og CAN-low ble identifisert ved inngangen i lastebilens FMS-system, hvor CANCrocodile ble klipset over og brukt for logging med CANedge1. OBD-porten var heller ikke tilgjengelig for CAN-tilkobling for Volvo. Med begrensede muligheter for å lete frem ledningene CAN-high og CAN-low ble løsningen å filme dashbordet for å registrere motordata.

Tilgjengelig CAN-data varierer mellom Scania og Mercedes-Benz. Med J1939-konverteringsfilen fra CSS Electronics ble 92 kanaler gjenkjent og transformert hos Scania, mot 75 kanaler hos Mercedes-Benz. Innholdet over tilgjengelige kanaler er i hovedsak likt hos fabrikantene, med enkelte unntak og variasjonsforskjeller. Ved filming av dashbordet blir viktige kanaler som turtall, gir og hastighet registrert, men motorens dreiemoment blir ikke kartlagt. Dreiemoment er en viktig parameter for utvikling av hastighetsmodellen, så under motbakker med filming av dashbordet gav sjåfør tilbakemelding på om maksimalt kraftpådrag ble krevd. Dreiemomentet kan også ved informasjon om turtall teoretisk bestemmes ved bruk av aktuell dreiemomentkurve.

9.2 Hastighetssammenlikning

Data i etterkant av forsøksturene ble sammenstilt for å sammenligne hastighet mellom CANedge1/dashbord og VBOX. Hastighet er en nøkkelparameter for viderutvikling av hastighetsmodellen, derfor var det viktig å avdekke eventuelle avvik i loggerdata mellom instrumentene fra kjøring med tunge kjøretøy. Av PGN-informasjonen til *WheelBasedVehicleSpeed* tilknyttet Scania, beregnes hastigheten enten fra bakdekkenes eller hele bakakselens rotasjonshastighet. Mercedes-Benz har kanalen *FrontAxleSpeed*, hvor hastigheten beregnes ved et gjennomsnitt fra de to dekkene på fremakselen. VBOX bruker posisjonsdata for å beregne hastighet.

Av alle loggerforsøkene var det tydelig at avviket mellom loggede hastigheter var størst for Scania. For Mercedes-Benz og Volvo var avviket knapt merkbart. Sammenstilling for et strekke fra andre forsøkstur med Scania er illustrert i figur 9.1.



Figur 9.1: Hastighetsprofiler fra loggerforsøk med Scania, VBOX og CANedge1

Av figur 9.1 er det gjennomgående at hastigheten logget fra VBOX er 2-3 km/t lavere enn fra CANedge1. Differansen minker noe når hastigheten er økende. Vogntogfører informerte om at dette hastighetsavviket har vist seg å ha mye med dekkene å gjøre, da avviket jevnt over er mindre etter omlegging til sommerdekk.

9.3 Teknologi for økonomisk kjøring

Lastebilfabrikanter har gjennom mange år jobbet med å utvikle teknologi som skal gi stadig mer drivstoffbesparende kjøring. Utbredt og felles for større fabrikanter er bruk av topografiske kartdata og posisjonsteknologi for å kartlegge veien forut. Basert på denne informasjonen velges en gir- og hastighetsstrategi for å minimere drivstofforbruk. Mye handler om å maksimere den utnyttede kinetiske energien til kjøretøyet ved å øke hastigheten før kjøretøyet ruller fritt i nøytral, hvor det ikke er kraftpådrag. For en motbakke fungerer systemet i praksis ved at kjøretøyet akselererer i bunnen av bakken og velger et gir som tilstrebes å holde til bakketoppen. Over bakketoppen ruller kjøretøyet fritt i nøytral, før et høygir velges for å akkumulere hastighet med lav motstand i nedoverbakke. Er bakken lang og hastigheten blir høy, initieres det en bremseprosess. Bremsene slippes opp tidlig for så å rulle lengst mulig i nøytral inn i det mindre bratte terrenget. På flatmark går den sykliske prosessen med kraftpådrag etterfulgt av fri rulling i nøytral.[Scania 2018]

Ved Scania er denne teknologien døpt *Scania Active Prediction* med *Pulse & Glide*, mens tilsvarende teknologi går under navnet *Predictive Powertrain Control*, PPC, ved Mercedes-Benz. Volvo opererer med *I-see*, *I-save* og *I-shift*.

De tre fabrikantene opererer alle med tre ulike former for kjøreprogram. Et økonomiprogram, et kraftprogram og en mellomting, et standardprogram. I økonomiprogrammene handler det om å kjøre billigst mulig ved å ligge i toppen av dreiemomentkurvene. Kraftprogrammene

drar gjerne litt lengre på girene og ofrer optimal økonomisk kjøring for å hente ut mer tilgjengelig effekt.

Prosjektgruppa diskuterte bruken av de ulike kjøreprogrammene med flåtestyringsansvarlige og sjåfører ved Børstad Transport. Det ble informert fra begge hold at det kjøres utelukkende på økonomiprogrammene for drivstoffbesparelse, med unntak av manøvrering ved lav hastighet i utfordrende høydeterrang typisk i forbindelse med kjøring på grusveier inn til enkelte gårder.

Dette er av interesse for utvikling av hastighetsmodellen. Et viktig aspekt er om det vil være realistisk å anta utnyttelse av maksimal tilgjengelig effekt, hvis det viser seg at svært få lastebiler kjører på et program som tillater dette. For betraktning av lastebiler i motbakker kommer i tillegg teknologien som fører til redusert kraftpådrag over bakketopper for drivstoffbesparelse.

10 Utvidelse av eksisterende hastighetsmodell

I kapittel 10 presenteres arbeidet som er blitt gjort i forbindelse med utvidelse av den eksisterende hastighetsmodellen til Børnes og Aakre (2011). Motbakkefremstilling og motstandskrefter har vært viktige fokusområder som legges frem, etterfulgt av de ulike delene av en modellspesifikk hastighetsmodell utviklet i programspråket Python. Videre følger resultatene av beregninger gjort med den nye modellspesifikke hastighetsmodellen satt opp mot den eksisterende hastighetsmodellen og reelle forsøk. Til slutt er begrensninger i den modellspesifikke hastighetsmodellen fremlagt.

10.1 Formålet med utvidelse

Formålet med en utvidet hastighetsmodell er å implementere motoreffekt og girvalg som avhenger av turtall, samt inkludere påvirkningen girskift har på gjeldene hastighet. Turtall varierer med hvilket gir det opereres i, samt aktuell hastighet. Ved minkende hastighet reduseres turtallet og et nytt lavere gir må velges. Det fører til et øyeblikk uten kraftpådrag under girskift, før et nytt gir med en ulik utvekslingskurve mellom hastighet og turtall velges.

Eksisterende modell benytter konstant effekt som er uavhengig av turtall, gir og hastighet. Det er derfor muligheter for å beskrive utnyttet effekt på en bedre måte. Motbakkefremstilling blir gjort ved oppdeling i et begrenset antall delstrekker. Dette kan gjøres mer nøyaktig ved å bruke kart- eller posisjonsdata uten begrensninger forbundet med oppdeling i delstrekker.

10.2 Motbakkefremstilling

En viktig forutsetning for en realistisk hastighetsmodell er korrekt fremstilling av motbakker som inngår i loggerforsøk, som videre blir brukt for videreutvikling og optimalisering av eksisterende modell. Det ble vurdert to ulike kilder til data for motbakkemodellering.

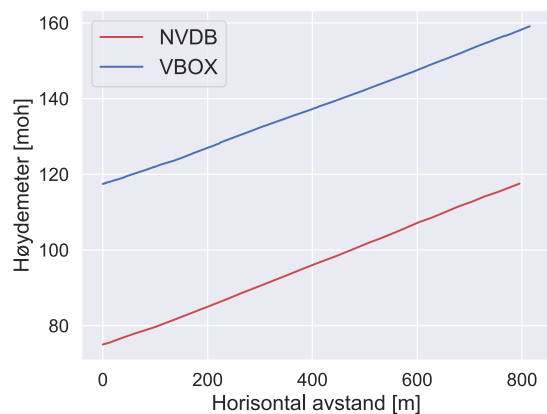
Den første kilden til data er eksisterende geometri tilhørende senterlinjer i NVDB. Statens vegvesen gav prosjektet en innføring i hvordan relevant høydedata skulle uthentes, men

kunne ikke gi et konkret svar angående geometrikvalitet. Metoden for motbakkemodellering ved bruk av data fra NVDB er beskrevet i kapittel 6.3.1.

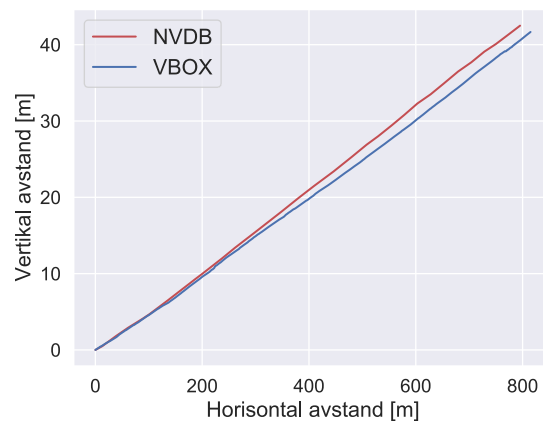
Den andre kilden er posisjonsdata logget direkte fra VBOX, som er beskrevet i kapittel 6.3.2. Z-koordinatene over en avstand for utvalgte stigninger i VBOX' loggerdata hentes inn i modellen for å fremstille motbakker.

For å vurdere de to metodene for fremstilling av motbakker ble begge tatt i bruk over et utvalg stigninger på Europaveg 6 etter loggerforsøkene beskrevet i kapittel 8.2 - 8.4. En gjentakende trend er at senterlinjen i NVDB er logget med langt lavere frekvens enn det ble gjort med VBOX under forsøkene. På tross av ulik loggerfrekvens er avvikene mellom fremstillingene små. I figur 10.1 og 10.2 er et utvalg av Okstadbakken på Europaveg 6 fremstilt med data fra NVDB og VBOX, før og etter et felles startpunkt er definert. Av figur 10.1 kommer det frem at høyderegreringene er svært forskjellige. VBOX' registreringer ligger i utgangspunktet i overkant av 40 meter høyere enn hva som er registrert i NVDB. Hva dette kommer av er ukjent, men det trenger nødvendigvis ikke være et hinder for selve motbakkefremstillingene, hvor det er stigningsendringen over et gitt strekke som er viktig å få modellert nøyaktig. I figur 10.2 er høyderegreringene sammenlignet fra et felles nullpunkt, som viser at data fra de to kildene stemmer godt overens. Sammenstilling av stigningsendringene fra de to datasettene er fremstilt i figur 10.3. Kartutsnitt med markerte start- og sluttpunkter er vedlagt i figur 10.4. Etter en avstand på 800 meter skiller det omtrent én høydemeter mellom de to metodene. I dette utvalget har VBOX 601 loggerpunkter mot 33 fra NVDB. Data fra utvalget av Okstadbakken er vedlagt i tabell 10.1.

Svingningene i VBOX' rådata er av en slik størrelse at det er vanskelig å avgjøre den faktiske stigningen i et gitt målepunkt. Det er den gjennomsnittlige stigningen over lengre strekker som er av betydning for hastighetsmodelleringen. Stigningsendringene for NVDB- og VBOX-data i figur 10.3 gir et grovt bilde over jevnheten til bakken og dataen. NVDB-grafen består av alle 33 loggepunktene, mens VBOX-dataen er redusert fra 601 punkter til 30 punkter. Hvert av de 30 punktene representerer gjennomsnittlig stigning for 20 og 20 verdier for å gi omtrent lik oppløsning for sammenligning med stigningsdata fra NVDB. Fullstendig kode for sammenligning av de to metodene for motbakkefremstilling finnes i vedlegg C.



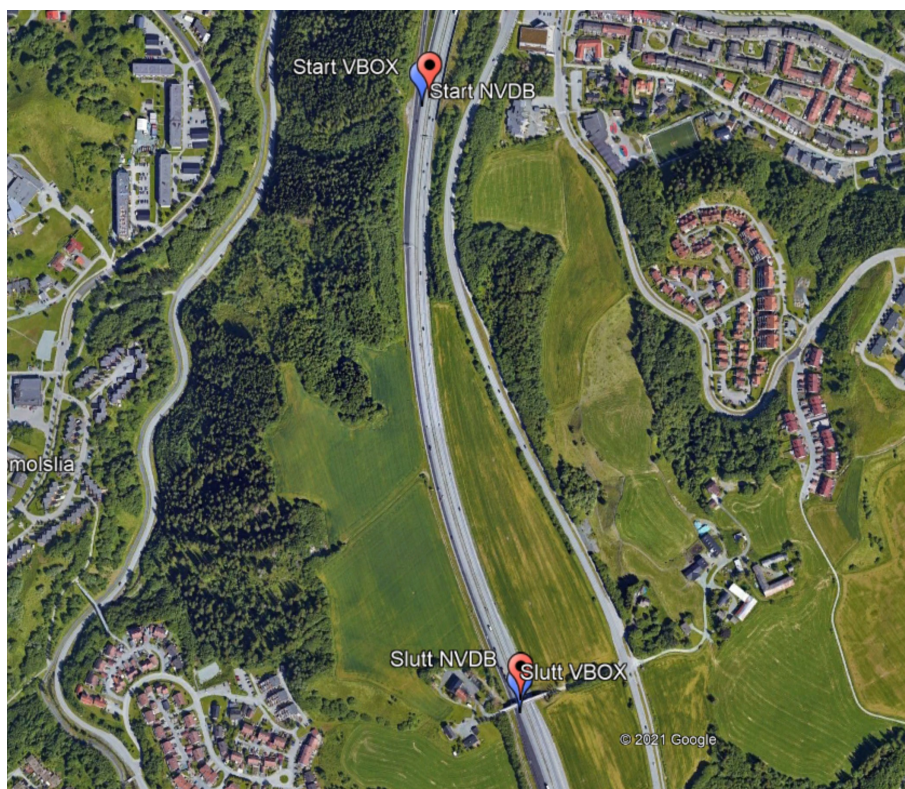
Figur 10.1: Høydeprofiler fremstilt fra NVDB- og VBOX-data



Figur 10.2: Felles nullpunkt for sammenligning av høydeprofiler fra NVDB- og VBOX-data



Figur 10.3: Sammenligning av stigning for NVDB- og VBOX-data



Figur 10.4: Kartutsnitt fra Google Earth over utvalg av Okstadbakken med start- og sluttkoordinat for loggerdata fra VBOX og NVDB (Skjerm bilde fra Google Earth)

	VBOX	NVDB
Startkoordinat	63°22.90368 N,10°22.48956 E	63°22.903015 N,10°22.498527 E
Sluttkoordinat	63°22.47818 N,10°22.64625 E	63°22.487073 N,10°22.647490 E
Loggepunkter	601	33
Lengde [m]	814,87	795,30
Høyde start [moh]	117,45	75,045
Høyde slutt [moh]	159,12	117,544
Høydeendring [m]	41,67	42,499
Stigning [%]	5,11	5,38

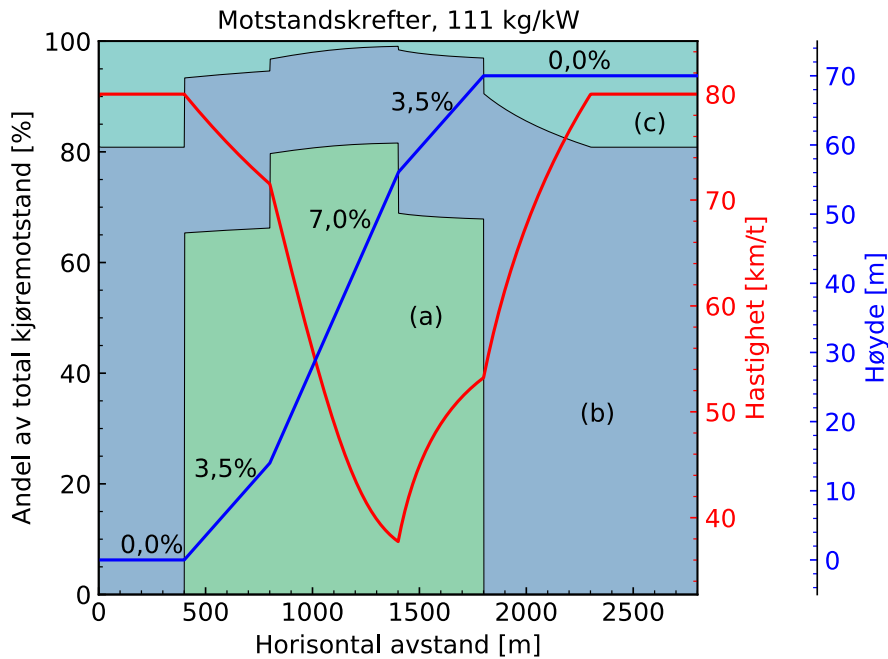
Tabell 10.1: Utvalg av Okstadbakken med stigningsdata fra VBOX og NVDB

Etter vurdering av de to metodene for motbakkefremstilling ble bruk av data fra VBOX foretrukket. Forskjellen mellom metodene er av en størrelse som vil ha liten påvirkning på resultatet av modellerte hastighetsprofiler over lengre strekker. VBOX' loggerfrekvens gjør at gradientendringer vil bli fanget opp raskt og gir høy oppløsning på fremstillingen. Bruk av NVDB-data bringer en inn en tredje datakilde som må sammenstilles med data fra VBOX og CANedge1. CSV-filen fra VBOX inneholder hastigheten til kjøretøyet fra forsøket og gjør det enkelt å sammenligne med modellert hastighet, da startpunkt og sluttpunkt blir definert av første og siste linje i filen. Modellen vil da alltid beregne over samme strekning som kjøretøyet ble logget over. Det presiseres at det kun sees på vertikal kurvatur innenfor motbakkefremstilling, horisontal kurvatur er ikke tatt med i beregningen.

10.3 Motstandskrefter

Kjøremotstanden bør modelleres på en måte som i størst mulig grad representerer virkeligheten når det jobbes mot å skape en realistisk hastighetsmodell. Det fokuseres på kjøremotstand i form av luftmotstand, rullemotstand og stigningsmotstand, som er de tre dominerende, kvantifiserbare motstandskreftene som virker på et kjøretøy i fart. Stigningsmotstand er avhengig av få parametere og er en veletablert og universell tolkning av denne typen motstandskraft. Ligning (10) presentert i kapittel 2.5.3 blir brukt i utvikling av hastighetsmodellen, med forenklingen $\sin(\alpha) = 0,01 \cdot p$, hvor p er stigningsgraden i prosent. For stigningsgrader på ≤ 10 % gir dette en feil på $\leq 0,5$ %.

Luftmotstand er presentert i kapittel 2.5.1 og gitt i ligning (6). Det er flere faktorer som inngår, og av ulike studier kan det være enkelte nyanseforskjeller i formelen for luftmotstand som blir benyttet for modellering. I dette prosjektet betraktes tunge kjøretøy i motbakke, som innebærer at hastigheten er relativt lav og stigningsmotstanden er høy. Dette er illustrert i figur 10.5. I grafen er bakken og kjøretøyet identisk med det som er brukt for å eksemplifisere bruken av eksisterende hastighetsmodell for forbikjøringsfelt i stigning, i Statens vegvesens håndbok V120 Premisser for geometrisk utforming av veger. [Vegdirektoratet, Statens vegvesen 2019b] Eksisterende hastighetsmodell for forbikjøringsfelt i stigning er beskrevet i kapittel 3.2. Bakken er delt opp i segmenter med ulik stigningsgrad, mens kjøretøyet har en vekt på 40 tonn og effekt på 360 kW, med 95 % utnyttelsesgrad. Fartsgrensen er 80 km/t. Bakgrunnsfargene i figur 10.5 indikerer fordeling av de tre ulike motstandskreftene ettersom kjøretøyet forflytter seg langs bakken med endret hastighet, hvor grønn (a) er stigningsmotstand, blå (b) er rullemotstand og turkis (c) er luftmotstand. Av grafen kommer det frem at luftmotstanden vil utgjøre en liten del av den totale kjøremotstanden i motbakke. Det er derfor vurdert som lite hensiktsmessig å undersøke om ytterlige tilpasninger kan og bør gjøres for å potensielt optimalisere formelen for luftmotstand. Ligning (6) blir brukt i utvikling av hastighetsmodellen.



Figur 10.5: Eksempelbakke med hastighet og kjøremotstandsfordeling

Rullemotstand er presentert i kapittel 2.5.2, med den generelle formelen i ligning (7). Rullefriksjonskoeffisienten (f_r) er avhengig av en rekke parametere og varierte tilnærminger til denne er blitt brukt i ulike studier. I enkelte tilfeller blir koeffisienten forenklet til en statisk snittverdi, uavhengig av ytre omgivelser under modellering.

Det er blitt gjort forskning på hva hastighet har å si for rullefriksjonskoeffisienten, ligning (8) og (9) er to muligheter som er blitt anvendt i modeller tidligere, hvor hastighet tas inn som en dynamisk parameter. I litteraturen er det konsensus om at økt hastighet gir økt rullemotstand. Ligning (8) ble vurdert som interessant, ettersom formelen tar inn både hastighet, dekkets overflateruhet og makrotekstur som dynamiske parametere. I det svenske studiet bak formelen ble det brukt forsøk med lastebiler for å komme frem til resultatet, som er relevant for dette prosjektet. [Hammarström mfl. 2012] Prosjektgruppa tok kontakt med Statens vegvesen, som hadde mulighet til å oversende data med de nyeste målingene av overflateruhet (iri) og makrotekstur (mpd) for ønsket strekning. Ligning (8) blir derfor brukt i utvikling av hastighetsmodellen. Rullefriksjonskoeffisienten vil ligge innenfor ekstremalverdier på 0,005 og 0,007 i hastigheter fra 30 km/t til 90 km/t. Dette er beregnet fra 5- og 95-persentiler av iri og mpd gitt for strekket Klett - Moholt, som ble kjørt under loggerforsøk på Europaveg 6. Rullefriksjonskoeffisienten for lastebiler ligger typisk mellom 0,006 og 0,01 i følge Bosch (2007), så brukt formel ser ut til å gi sannsynlige verdier.

Ved beregning av rullefriksjonskoeffisient for motbakker på Europaveg 6 blir det i den videreutviklede hastighetsmodellen brukt snittverdier for *iri* og *mpd* på henholdsvis 1,2 m/km og 1,3 mm. Snittverdier er beregnet fra nevnte datautvalg på Europaveg 6 fra Klett til Moholt. Avviket i hastighetsmodellen ved å benytte snittverdier kontra punktmålinger for *iri* og *mpd* er i praksis neglisjerbare, derfor ble det vurdert som unødvendig å implementere koden som tar inn hele lister med punktmålinger for ny beregning i hvert enkelt tidssteg. Koden for å få til dette ble likevel utviklet og finnes i vedlegg D.

10.4 Modellspesifikk hastighetsmodell

Utgangspunktet for følgende modellspesifikk hastighetsmodell er hastighetsmodellen utviklet i Excel av Børnes og Aakre (2011) som er beskrevet i kapittel 3.1. Basisen er lik, med beregning av akselerasjon og hastighet med en gitt stegfrekvens, basert på drivkrefter og motstandskrefter. For videreutvikling av hastighetsmodellen med fokus på motbakkefremstilling, turtall, gir og effekt ble det bestemt at programspråket Python trolig ville være mer effektivt enn Excel for å gjøre mer avanserte beregninger. Fullstendig kode for modellspesifikk hastighetsmodell er vedlagt i vedlegg E.

10.4.1 Sammenheng mellom hastighet og turtall

For de ulike lastebilmodellene er det i den modellspesifikke hastighetsmodellen laget lister med forholdstall mellom turtall og hastighet for de fem høyeste girene. Dette forholdstallet vil være konstant i hvert enkelt gir. Eksempelvis ved økt hastighet i et gir øker turtallet tilsvarende med en gitt faktor. Listene er gitt i tabell 10.2. Forholdstallene er basert på girutvekslingskurvene for lastebilmodellene presentert i kapittel 8.2 til 8.4. Grunnen til at beregningen ikke blir gjort for lavere gir er at de ikke blir initiert under noen av forsøkene eller under modellberegningene i motbakke, da hastigheten ikke blir tilstrekkelig lav.

Gir	Scania R580	Mercedes-Benz Arocs	Volvo FH 540
12	14.10	16.30	15.52
11	17.80	21.10	19.78
10	21.20	26.20	25.31
9	26.60	33.95	32.25
8	35.00	43.40	41.92

Tabell 10.2: Lastebilmodellenes forholdstall mellom turtall og hastighet for toppgirene

10.4.2 Girvalg

I hastighetsmodellen er de operasjonelle turtallsområdene for hver enkelt lastebilmodell før et girskift blir initiert definert på bakgrunn av dreiemomentkurvene presentert i kapittel 8.2 til 8.4, samt erfaring fra forsøksturene. Turtallsområdene er listet i tabell 10.3. Ved økonomisk kjøring etterstreber lastebilen å ligge i toppen av dreiemomentkurven for å minimere forbruk. Dette stemmer godt overens med observasjonene fra forsøksturene. Mercedes-Benz ble kjørt på standardprogram, noe som kan forklare et noe høyere operasjonelt turtallsområde i forhold til Scania og Volvo, som ble kjørt på økonomiprogram. I

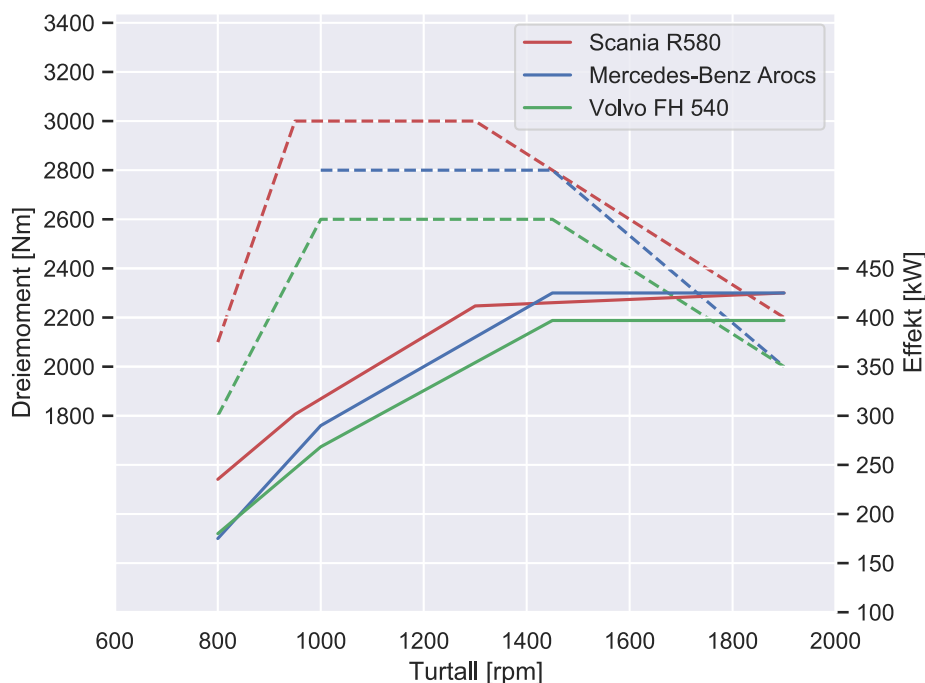
realiteten er det flere faktorer som spiller inn når de nyere adaptive automatiske girkassene gjør girvalg, dette er kommentert i kapittel 9.3. Under girskift er det i hastighetsmodellen som hovedregel lagt inn ett sekund hvor lastebilen ikke har kraftpådrag, for å simulere et reelt girskift. En effekt av implementering av girskift i hastighetsmodellen er valg av gjeldende gir når beregningen starter.

Scania R580	Mercedes-Benz Arocs	Volvo FH 540
1000-1400	1200-1600	1000-1450

Tabell 10.3: Lastebilmodellenes operasjonelle turtallsområder

10.4.3 Variabel motoreffekt

Basert på effektkurvene for hver lastebilmodell presentert i kapittel 8.2 til 8.4, er det i hastighetsmodellen laget segmenter som går på forholdet mellom turtall og effekt langs de enkelte effektkurvene. Segmentene er illustrert i figur 10.6. For segmentene blir det beregnet stigningstall som beskriver utviklingen langs effektkurven når turtallet øker eller minker. Tilgjengelig effekt vil dermed variere med turtallet. V8-motoren til Scania har høyere dreiemoment enn de 6-sylindrede motorene til Mercedes-Benz og Volvo, og vil kunne operere på et lavere turtall samtidig som den leverer samme effekt.



Figur 10.6: Konstruerte moment- og effektkurver for lastebilene benyttet under forsøk

10.4.4 Fremstilling

Den modellspesifikke hastighetsmodellen beregner videre i kapittelet hastighetsprofiler for et utvalg motbakker som ble kjørt under loggerforsøkene, for de aktuelle lastebilmodellene fra Scania, Mercedes-Benz og Volvo. Beregningsfrekvensen er satt til hvert hundredels sekund. Det er minimal forskjell i resultatene mellom en beregningsfrekvens på hvert tidels og hundredels sekund, men koden bruker i praksis like lang tid på å utføre beregningen. Ved beregningsfrekvens på opp mot ett sekund kan enkelte av de ulike segmentene motbakken er bygget opp av unnslippe beregning, som ikke er ønsket. Oversikt over de ulike grafene med motbakkeinformasjon er listet i tabell 10.4. I grafene representerer den skarpe blå linjen motbakken, med vertikal avstand langs høyre y-akse og horisontal avstand langs x-aksen. De tre andre linjene som følger samme form representerer hastigheter og tilhører venstre y-akse med beskrivelse i km/t, mens x-aksen er horisontal avstand i likhet med linjen for motbakke.

Oransje stiplet linje er beregnet hastighetsprofil med den eksisterende “gamle” modellen til Børnes og Aakre. Grønn linje er målt hastighetprofil fra VBOX under forsøkene, mens rød linje er beregnet hastighetsprofil med ny modellspesifikk hastighetsmodell. Tidspunkt for girskift er markert med vertikale linjer, grå for gammel hastighetsmodell og turkis for ny hastighetsmodell. Gjeldende gir før og etter girskift er illustrert i boksene over de vertikale linjene.

Et viktig moment er at den gamle hastighetsmodellen til Børnes og Aakre blir beregnet med lik motbakkedata fra VBOX, samt beregningsfrekvens som den nye hastighetsmodellen. Dette gjør at hastighetsprofilen blir mer detaljert enn den eksisterende modellen beskrevet i kapittel 3.1, som deler opp motbakker i inntil 5 delstrekker.

Lastebilmodell	Figur	Område	Vegreferanse fra - til	Snittgradient
Scania R580	10.7	Okstad	EV6 S75D1 m2093-140	4,66 %
Scania R580	10.8	Sveberg	EV6 S78D1 m3235-1780	3,59 %
Scania R580	10.9	Vuddu	EV6 S81D1 m6952-5434	5,36 %
Mercedes-Benz Arocs	10.10	Okstad	EV6 S75D1 m1801-350	4,82%
Mercedes-Benz Arocs	10.11	Moholt	EV6 S76D1 m2702-3369	6,79%
Mercedes-Benz Arocs	10.12	Klett	EV6 S73D1 m1202-3029	5,17%
Volvo FH 540	10.13	Okstad	EV6 S75D1 m2168 - EV6 S74D1 m3370	4,18%

Tabell 10.4: Motbakkeinformasjon ved beregning med modellspesifikk hastighetsmodell

For å sammenligne resultatene fra eksisterende og ny hastighetsmodell i forhold til reelt hastighetsprofil under forsøktsturene, er det valgt å bruke GEH-verdier. GEH etter Geoffrey E. Havers, er et statistisk estimat som i hovedsak brukes i trafikkmodellering for å angi grad av samvariasjon mellom beregnede og observerte verdier. Den matematiske formelen kategoriseres ikke som en genuin statistisk test, men er et nyttig verktøy for å tallfeste likhet mellom datasett.

Formelen for gjennomsnittlig GEH-verdi er presentert i ligning 17.

$$\overline{GEH} = \frac{1}{n} \sum_{i=1}^n \sqrt{\frac{2(E_i - V_i)^2}{E_i - V_i}} \quad (17)$$

E og V i ligningen beskriver henholdsvis modellert og reell hastighet i denne oppgaven. For hvert modellert hastighetsprofil er det beregnet gjennomsnittlig GEH-verdi over modelleringsområdet, i forhold til reelt hastighetsprofil. Verdiene er deretter ganget med 100 for å gi tall i størrelsesorden mellom 0 og 10 for enkel skilnad. Av formelen som er presentert, impliserer lavere GEH-verdier mindre forskjell mellom hastighetsprofilene. To identiske hastighetsprofil vil gi en GEH-verdi på 0.

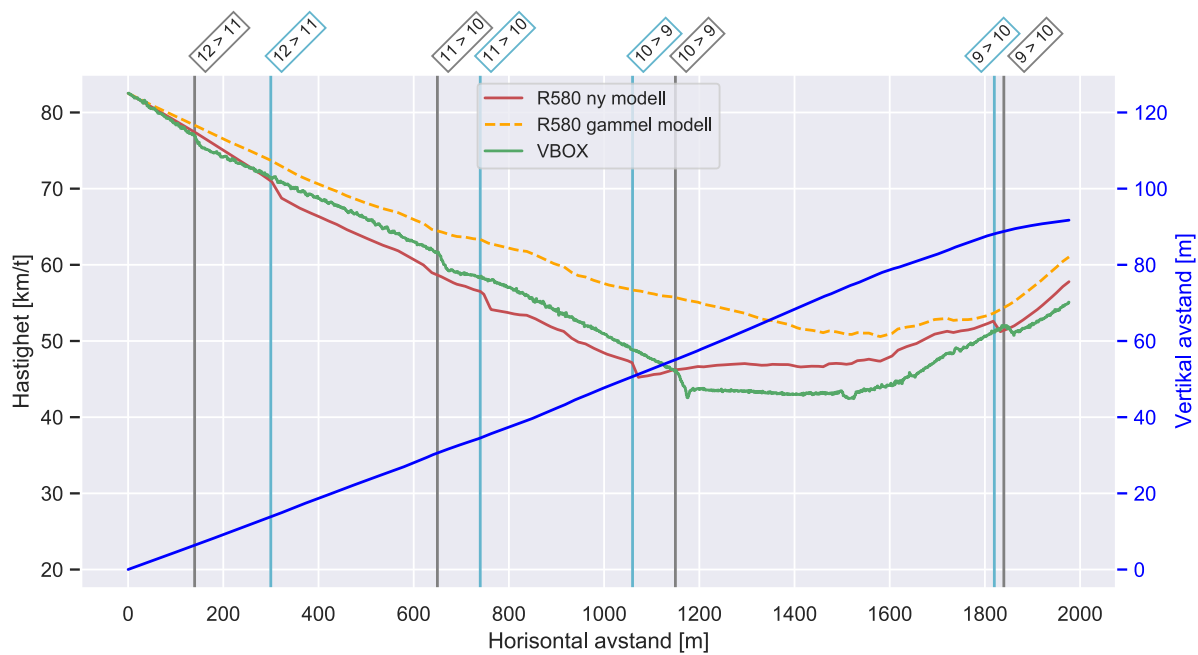
10.4.5 Scania

For de tre modellerte hastighetsprofilene sammen med feltforsøkene tilhørende vogntoget Scania R580, starter kjøretøyet i henholdsvis 12., 10., og 11. gir. Vogntogets totalvekt er 50 tonn, med en maksimal motoreffekt på 427 kW. Frontareal og dragkoeffisient er satt til henholdsvis 8 m² og 0,6.

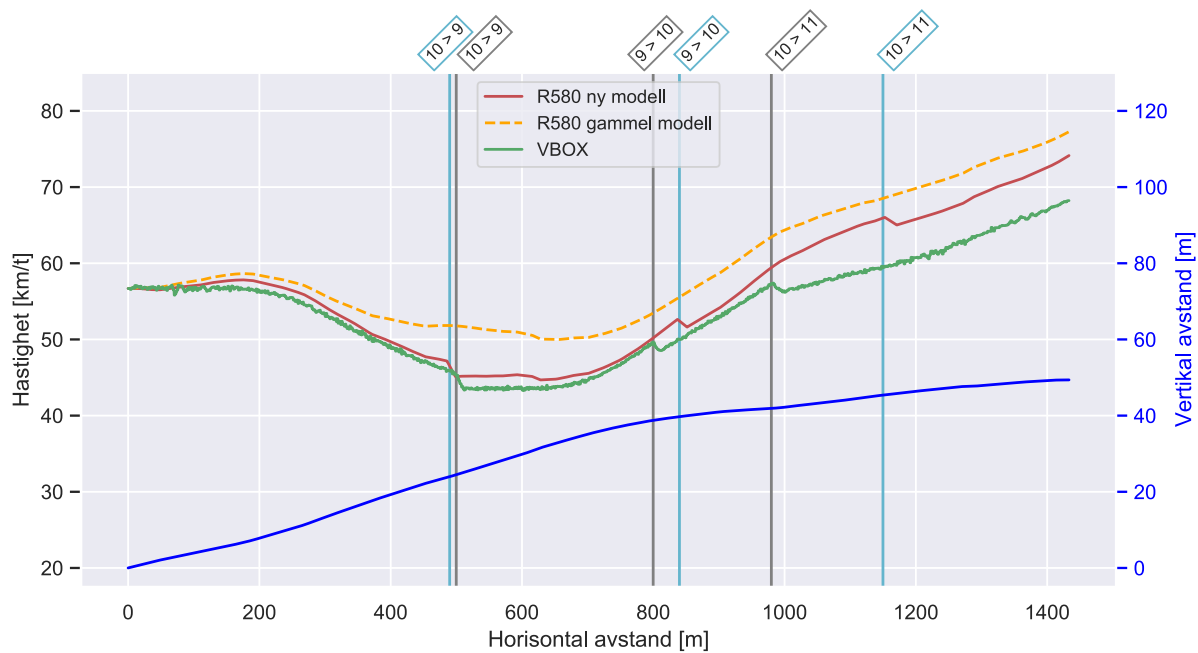
Under modellering i Okstadbakken i figur 10.7 blir det gjort like mange girskift i den nye modellen som under forsøksturen, og av tabell 10.5 kommer det frem at total tid brukt på å gjennomføre stigningen er lik mellom ny modell og under forsøksturen. Laveste nådde hastighet er lavere under forsøksturen i forhold til ny modellert hastighet, mens den gamle modellen beregner den høyeste minimumshastigheten og bruker kortest tid på å gjennomføre stigningen.

For modellering i stigningen på Sveberg i figur 10.8 beregnes det også like mange girskift i den nye modellen som under forsøksturen. Her stemmer tidspunktene for nedgiring bedre overens med forsøksturen sammenlignet med Okstadbakken. Ny modell bruker tre sekunder mindre på å gjennomføre stigningen sammenlignet med forsøksturen. Laveste nådde hastighet stemmer godt overens med ny modell og forsøksturen, med en differanse på under 2 km/t. Igjen bruker den gamle modellen kortest tid på å gjennomføre stigningen og har den høyeste minimumshastigheten.

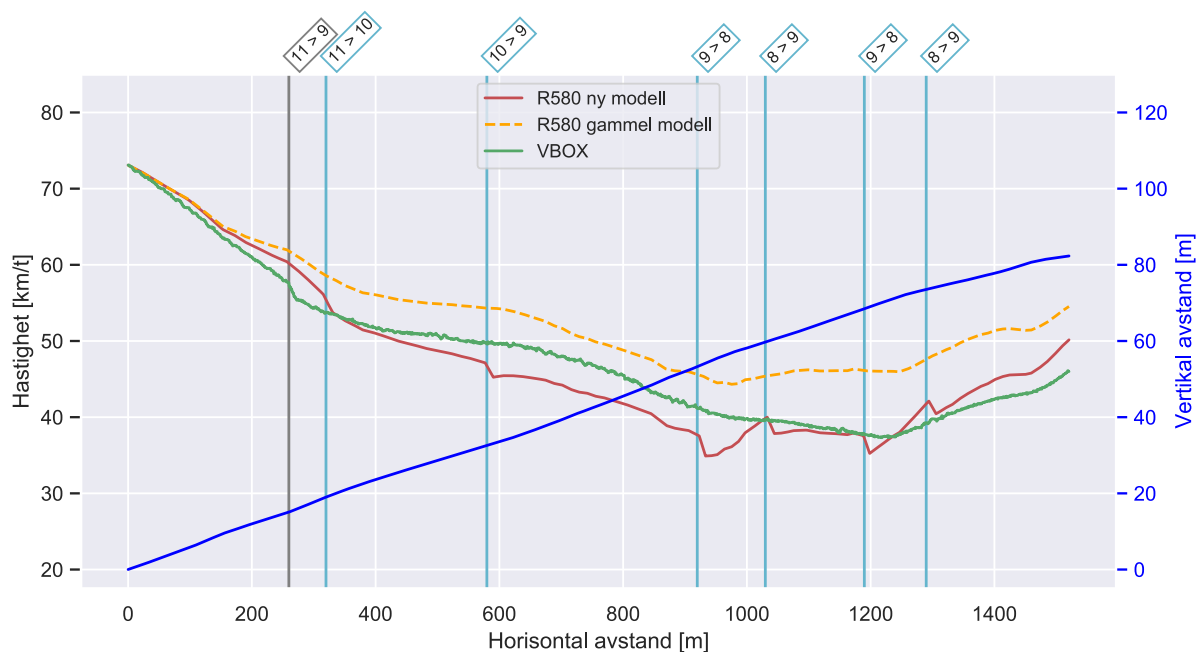
Under modellering av stigningen i Vuddudalen i figur 10.9 blir det gjort langt flere girskift i den nye modellen sammenlignet med forsøksturen. Modellen opererer over et grenseområde for girskift mot slutten, slik at flere girskift mellom 8. og 9. gir blir gjort før hastigheten blir høy nok til å unngå nedgiring. Under forsøksturen ble det gjort én enkel nedgiring fra 11. til 9. gir, som ble holdt under resten av stigningen. På tross av ulik girstrategi er total tid på å gjennomføre stigningen relativt lik mellom den nye hastighetsmodellen og forsøksturen, hvor det ble brukt fire sekunder kortere tid under forsøksturen. Her er laveste nådde hastighet i overkant av 2 km/t lavere for den nye hastighetsmodellen sammenlignet med forsøksturen. Den gamle modellen har høyest minimumshastighet og bruker kortest tid på å gjennomføre stigningen.



Figur 10.7: Spesifikk hastighetsmodell, Okstad, Scania R580



Figur 10.8: Spesifikk hastighetsmodell, Sveberg, Scania R580



Figur 10.9: Spesifikk hastighetsmodell, Vuddu, Scania R580

Hastighetsmodell	Tidsbruk	Laveste hastighet	GEH-verdi
Okstad			
Gammel	1m 58s	50,6 km/t	7,86
Ny	2m 10s	45,2 km/t	3,32
VBOX	2m 10s	42,4 km/t	-
Sveberg			
Gammel	1m 18s	50,0 km/t	7,09
Ny	1m 34s	44,7 km/t	2,54
VBOX	1m 37s	43,3 km/t	-
Vuddu			
Gammel	1m 44s	44,3 km/t	8,21
Ny	2m 00s	34,9 km/t	3,79
VBOX	1m 56s	37,3 km/t	-

Tabell 10.5: Tids- og hastighetsdata med GEH-verdier tilhørende figur 10.7, 10.8 og 10.9

10.4.6 Mercedes-Benz

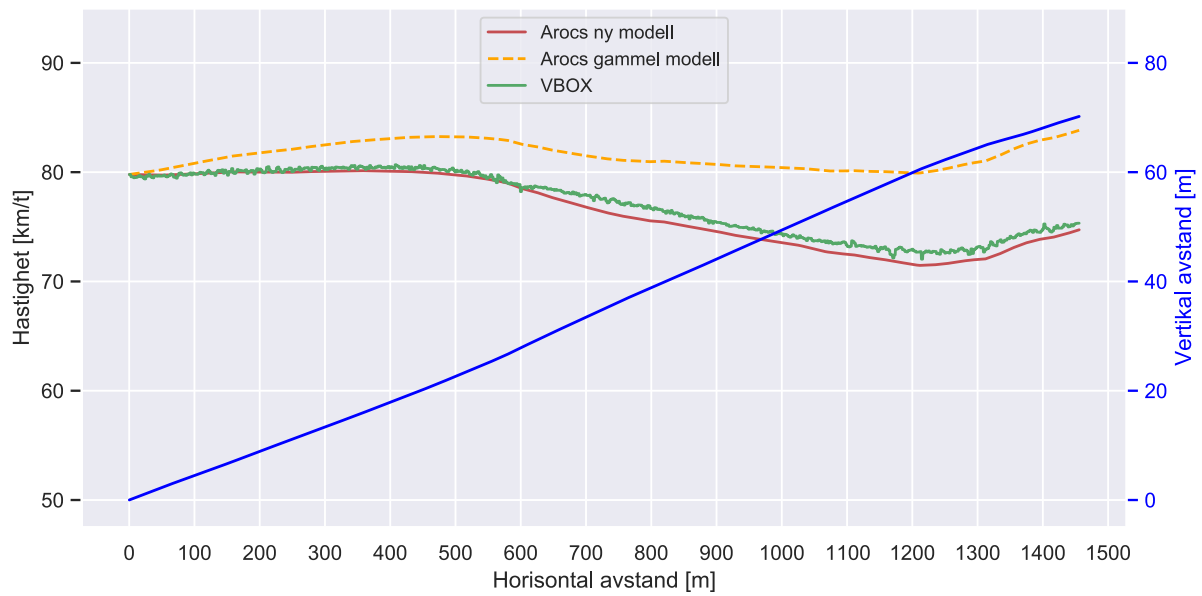
For de tre modellerte hastighetsprofilene og feltforsøkene tilhørende lastebilen Mercedes-Benz Arocs, starter kjøretøyet alltid i 12. gir. Lastebilens vekt er 28 tonn, med en maksimal motoreffekt på 425 kW. Under feltforsøkene ble det også kjørt med halv last, men under disse forsøkene ble totalvekten så lav i sammenheng med motorstørrelsen at lastebilen hadde få problemer med å holde hastigheten i motbakkene. Det blir derfor sett på modelleringssituasjoner med full last. Fra Mercedes-Benz ble det gitt spesifikk informasjon om at produktet av frontareal og dragkoeffisient for lastebilmodellen er 7,13. Denne verdien blir derfor brukt under modellering i formelen for luftmotstand.

Under modellering i Okstadbakken i figur 10.10 blir ikke hastigheten lav nok til å gjennomføre girskift verken i den nye modellen og under feltforsøket. Hastighetsprofilen generert i den nye modellen stemmer godt overens med feltforsøket, med kun en halv km/t differanse i laveste beregnede hastighet av tabell 10.6. De sammenfallende hastighetsprofilene gjenspeiles også i tidsforbruk på stigningen, med en differanse på ett sekund. Den gamle modellen har en langt høyere beregnet minimumshastighet, med rundt 8 km/t mer enn under feltforsøket.

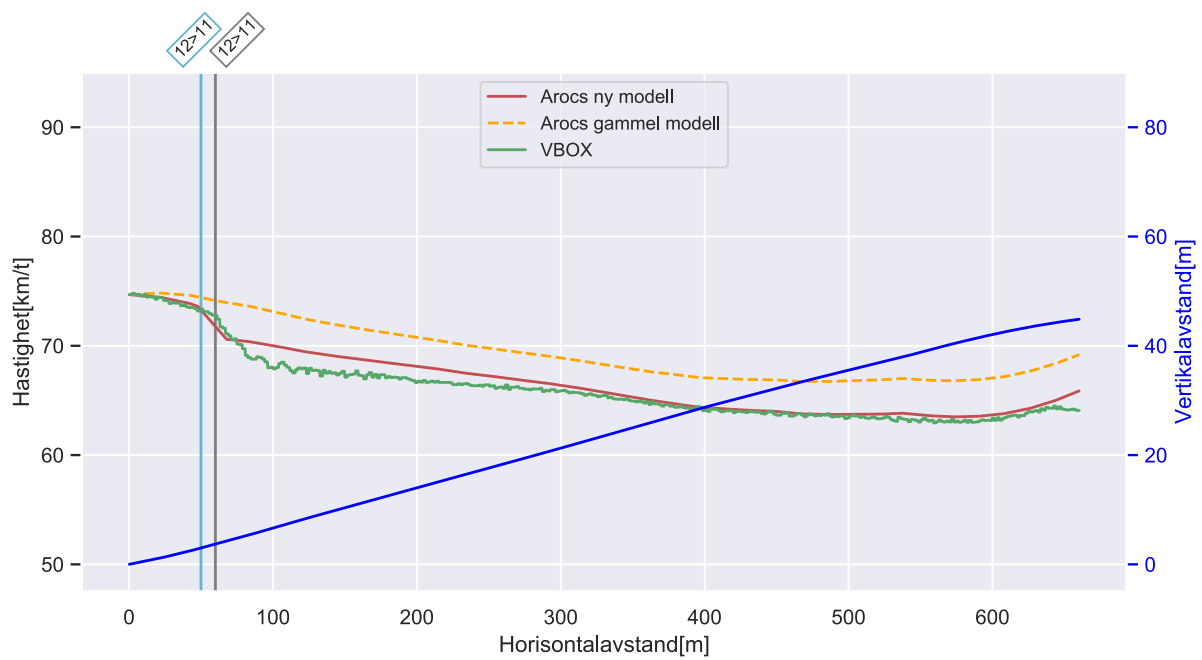
Under modellering i Moholtbakken i figur 10.11 gjennomføres det et girskift fra 12. til 11. gir i den nye modellen og under feltforsøket. Tidspunktet for girskiftet sammenfaller godt i de to hastighetsprofilene. Laveste beregnede hastighet og tidsforbruk på stigningen er tilnærmet lik. Den gamle modellen har også her en høyere beregnet minimumshastighet, på rundt 4 km/t.

Modelleringen av hastigheten i Klettbakken i figur 10.12 samsvarer godt med feltforsøket og første nedgiring skjer nesten identisk med virkeligheten. Fra 920 meter til 1050 meter går Europaveg 6 i kulvert og VBOX mistet signalet fra satellittene under feltforsøket. Dette resulterte at hastigheten ble registrert som null i en kort periode. For å gjøre hastighetsprofilen mer oversiktlig er det valgt å sette hastigheten til 72 km/t for dette segmentet, basert på hastigheten før og etter signaltapet. Signaltapet forstyrret følgelig også motbakkeprofilen, som kommer til uttrykk i form av en liten dump i hastighetsprofilen til ny og gammel modell rundt 1000 meter. Her ønsket i utgangspunktet den nye modellen å gjennomføre en nedgiring, men dette ble den manuelt hindret i å gjøre, ettersom grunnlaget for nedgiringen ikke ville vært riktig. Den totale påvirkningen signaltapet har på modellerte hastighetsprofil er svært liten.

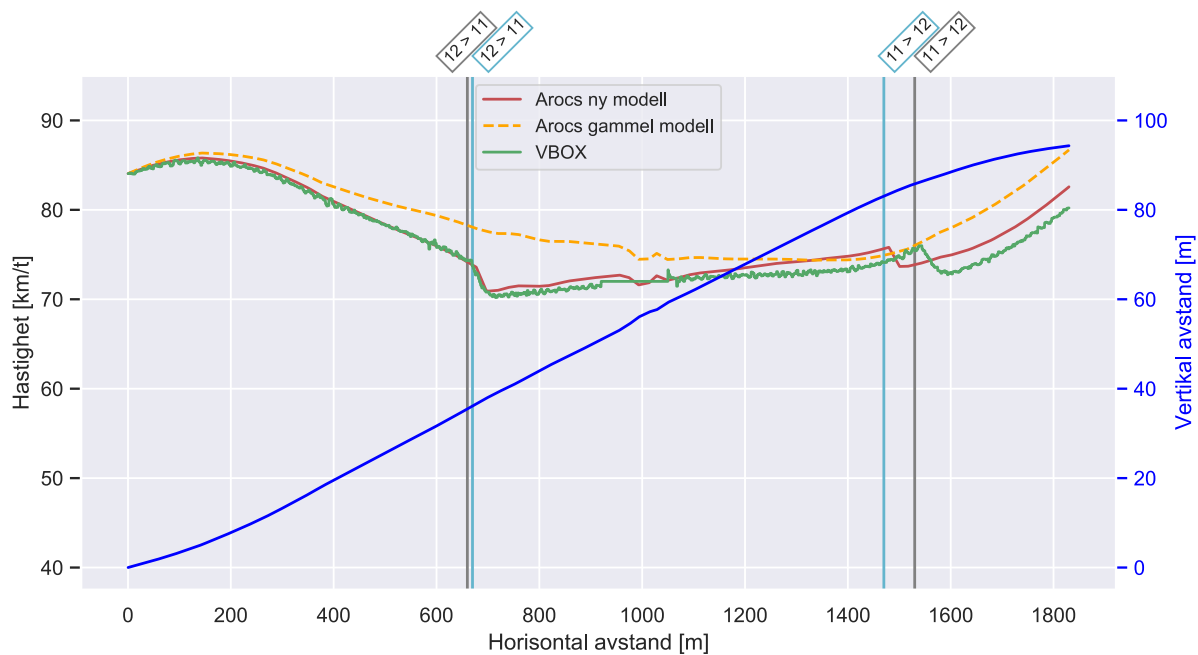
Tidspunkt for girskift i Klettbakken er godt sammenfallende for den nye modellen og feltforsøket, både for nedgiringen og oppgiringen mellom 12. og 11. gir. Laveste beregnede hastighet og tidsforbruk på stigningen er også her som i Okstadbakken og Moholtbakken, tilnærmet lik. Igjen har den gamle modellen en høyere beregnet minimumshastighet, med forskjell på rundt 4 km/t.



Figur 10.10: Spesifikk hastighetsmodell, Okstad, Mercedes-Benz Arocs



Figur 10.11: Spesifikk hastighetsmodell, Moholt, Mercedes-Benz Arocs



Figur 10.12: Spesifikk hastighetsmodell, Klett, Mercedes-Benz Arocs

Hastighetsmodell	Tidsbruk	Laveste hastighet	GEH-verdi
Okstad			
Gammel	1m 4s	79,8 km/t	3,99
Ny	1m 9s	71,5 km/t	0,71
VBOX	1m 8s	72,0 km/t	-
Moholt			
Gammel	34s	66,7 km/t	3,48
Ny	36s	63,5 km/t	0,66
VBOX	36s	63,0 km/t	-
Klett			
Gammel	1m 23s	74,4 km/t	3,22
Ny	1m 26s	70,9 km/t	1,41
VBOX	1m 27s	70,2 km/t	-

Tabell 10.6: Tids- og hastighetsdata med GEH-verdier tilhørende figur 10.10, 10.11 og 10.12

10.4.7 Volvo

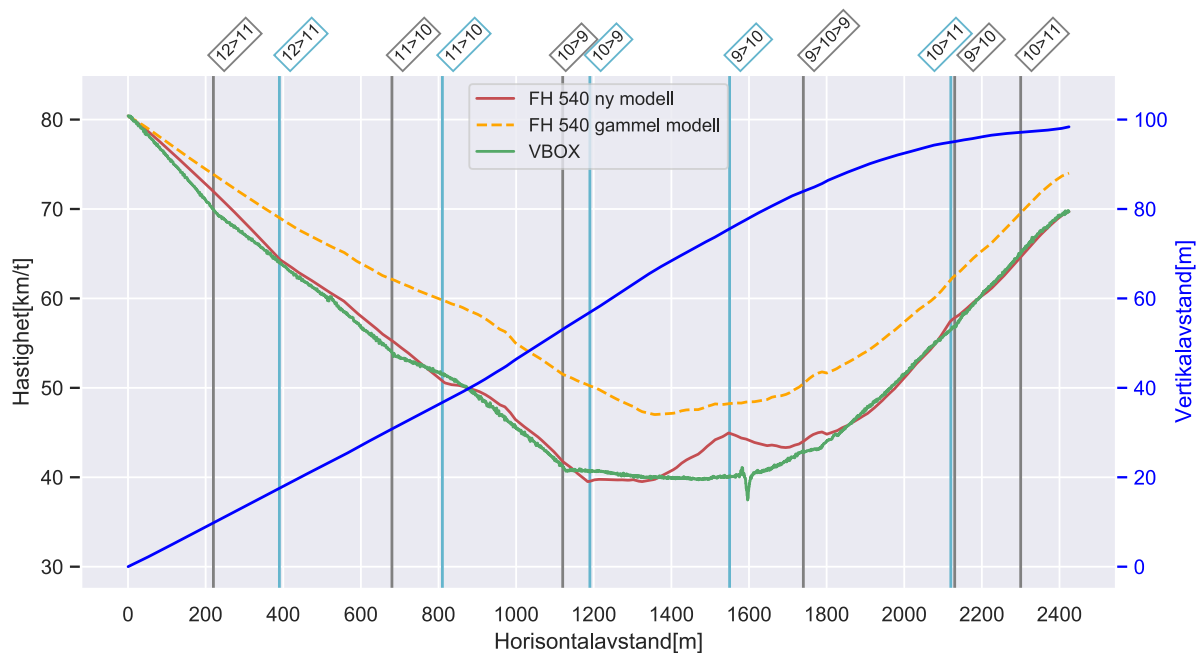
For Volvo FH 540 er kun ett hastighetsprofil modellert, i Okstadbakken. Vogntoget veier 50 tonn og starter i 12. gir, med en maksimal motoreffekt på 397 kW. Frontareal og dragkoeffisient er satt til henholdsvis 8 m^2 og 0,6. Volvo fremhever sin teknologi for giring gjennom *I-shift*, med perfekt timede girskift. Det viser seg at ved hastighetsmodellering i Okstadbakken gir det bedre samsvar mellom nytt modellert hastighetsprofil og forsøksturen ved å ekskludere øyeblikket uten kraftpådrag under girskift fullstendig. Dette er derfor gjort i hastighetsprofilen modellert i figur 10.13.

Fra 1400 meter avtar stigningsgraden og turtallet øker, derav fartsøkningen i den nye modellen. Denne fartsøkningen fant ikke sted under forsøksturen. Trolig ville kjøretøyet i virkeligheten beholde et lavere turtall og unngikk dette kraftpådraget. Dette er vanskelig å kunne bekrefte, da CAN-data ikke ble logget for forsøksturen. Ny modell girer opp til 10. gir ved 1550 meter da turtallet når 1450 omdreininger. I etterkant av dette girskiftet faller hastigheten og nærmer seg logget hastighet fra forsøksturen. Kraftpådraget til den nye modellen kan da sees på som unødvendig bruk av energi.

Ved 1750 meter valgte vogntoget under forsøksturen å gire opp til 10. gir, men angret seg umiddelbart i form av påfølgende nedgiring. En mulig forklaring kan være en uventet gradientendring, ettersom kjøretøyet var uhindret av trafikk og ble kjørt på kruskontroll. Høydedataen viser at plutselig faller stigningsgraden fra 4,5 % til 3 %, for så å brått gå opp til 5 % og så ned igjen til ca. 4,5 %, alt i løpet av en distanse på under 100 meter. Det ser ikke ut til at dette ble forutsett av Volvos intelligente kjøresystem beskrevet i kapittel 9.3. Lengden på dumpen er muligens for kort til å undergå en isolert vurdering fra systemet.

Generelt samsvarer tidspunkt for giring relativt godt mellom ny modell og forsøksturen frem til nevnte utfordringer oppstår. Fra 1550 meter blir det ulike girstrategier, hvor den nye modellen er tidligere ute med oppgiringen i forhold til hva som ble gjort under forsøksturen. Hastighetsprofilene er på tross av dette svært sammenfallende.

Av tabell 10.7 bruker den nye modellen et sekund kortere tid på å gjennomføre stigningen, sammenlignet med forsøksturen. Laveste nådde hastighet er i overkant av 2 km/t høyere for den nye modellen i forhold til forsøksturen. Sannsynligvis er laveste nådde hastighet under forsøksturen rundt 40 km/t, da laveste registrerte hastighet er i forbindelse med et fall under dårlige signalforhold. Den gamle modellen bruker langt kortere tid på å gjennomføre stigningen, med 20 sekunder mindre enn under forsøksturen. Laveste nådde hastighet for den gamle modellen er i underkant av 10 km/t høyere i forhold til forsøksturen.



Figur 10.13: Spesifikk hastighetsmodell, Okstad, Volvo FH 540

Hastighetsmodell	Tidsbruk	Laveste hastighet	GEH-verdi
Gammel	2m 29s	47,0 km/t	9,45
Ny	2m 48s	39,5 km/t	1,43
VBOX	2m 49s	40.0 km/t	-

Tabell 10.7: Tids- og hastighetsdata med GEH-verdier tilhørende figur 10.13

10.4.8 Oppsummering av modellberegninger

Gjentakende for alle modellberegningene er at eksisterende modell predikerer hastighetsprofil som er for optimistiske, med kortere tidsbruk på å gjennomføre stigninger, samt høyere minimumshastighet enn under forsøk. Dette gjenspeiles i GEH-verdiene hvor eksisterende modell utelukkende har markant høyere verdier sammenlignet med ny modell. Det tyder på at den generelle antakelsen i eksisterende modell, med høy utnyttelsesgrad av konstant maksimal motoreffekt under modellering samsvarer mindre godt med virkeligheten.

For den nye modellspesifikke hastighetsmodellen er det generelt godt samsvar med forsøksturene for tidsbruk og laveste nådde hastighet. Tidspunkt for giring og girvalg stemmer imidlertid ikke alltid like godt. Det viser seg vanskelig å konstruere en modellspesifikk hastighetsmodell som timer girskift slik som i virkeligheten, ettersom girvalg i dagens automatiske girkasser er langt mer komplisert enn å kun være turtallavhengig. De konstruerte segmentene som går på forholdet mellom turtall og effekt for hver enkelt lastebilmodell presentert i kapittel 10.4.3, er også en forenkling av de respektive effektdiagrammene. Beskrivelse av forholdet mellom turtall og effekt som eksakte kurver for de ulike lastebilmodellene vil trolig bidra til et jevnere hastighetsprofil slik som under forsøksturene, med færre tilfeller av opp- og nedgiring som eksempelvis finner sted i figur 10.9.

Oppførsel under selve girskiftene har også vist seg å kunne variere mellom fabrikanter. Ved å betrakte Volvos hastighetsprofil fra forsøksstur i figur 10.13, er det tydelig at effekttapet under girskift er langt mindre enn for Scania og Mercedes-Benz i figurene 10.7 - 10.12.

Ujevnheter i motbakkefremstillinger gir tydelige utslag på modellert hastighetsprofil, men hastighetsprofilene fra ny modell og forsøksturene blir raskt sammenfallende igjen. Det tyder på at lokal måleusikkerhet har liten påvirkning på totalt modellert hastighetsprofil og tidsbruk over ulike stigninger.

10.5 Begrensninger i modellen

Den eksisterende modellen til Børnes og Aakre er blitt utvidet til en modellspesifikk hastighetsmodell på bakgrunn av tekniske spesifikasjoner og forsøkssturer med lastebilmodeller fra fabrikanterne Scania, Mercedes-Benz og Volvo. Hastighetsmodellen beregner fartsprofil basert på kjøretøy og motbakke. Variabel motoreffekt og girskift er blant nye implementasjoner i hastighetsmodellen, sammen med variabel rullemotstand og mer nøyaktig motbakkefremstilling. Utover de fremlagte utvidelsene og optimaliseringene i hastighetsmodellen, er det flere aspekter knyttet til modelleringen som ikke er inkludert eller kan gjøres mer nøyaktig. På sikt er en mulighet å kombinere flere studier som fokuserer på ulike modelleringsområder for å optimalisere hastighetsmodellen ytterligere i forhold til virkeligheten. Kapittelet tar videre for seg ulike områder som den modellspesifikke hastighetsmodellen ikke tar hensyn til, med potensielle innvirkninger på resultatet.

10.5.1 Horisontalgeometri

Vegens horisontalgeometri er ikke inkludert i hastighetsmodellens beregningsgrunnlag. På tross av at vertikalgeometri vil ha langt større påvirkning på hastighetsprofilen, er det uvisst om det er realistisk å neglisjere horisontalgeometriens påvirkning fullstendig. Krappere svinger fører til økt sentripetalakselerasjon, som fordrer at mer krefter må brukes på å holde kjøretøyet på vegbanen gjennom svingene.

10.5.2 Kjøremotstand

Realistiske modelleringer av motstandskrefter er kompliserte øvelser i seg selv. I hastighetsmodellen er det tatt utgangspunkt i formler for kjøremotstand som er blitt brukt i lignende studier tidligere. Formelen for rullemotstand er utvidet fra eksisterende modell ved avhengighet av hastighet og vegens overflatetekstur. Det blir brukt en fast statisk dragkoeffisient i formelen for luftmotstand, uavhengig av kjøretøy for Scania og Volvo. Denne parameteren vil i realiteten variere mellom kjøretøy avhengig av utforming. Vindpåvirkningen hensynstas kun i kjøretøyets fartsretning, som er en forenkling. I ytterste konsekvens vil sterk vind vinkelrett på fartsetningen ikke ha effekt på modellert hastighetsprofil, som trolig ikke stemmer overens med virkeligheten.

10.5.3 Ytre omgivelser

Ytre omgivelser som vær, føre og trafikkforhold hensynstas ikke i hastighetsmodellen. Rullefriksjonskoeffisienten er modellert uavhengig av føre, men øker i realiteten ved fremmedlegemer i vegen som eksempelvis regn, snø og slaps. Hastighetsmodellen gjelder kun i situasjoner hvor kjøretøyet er uhindret av all type trafikk. Trafikkforhold er komplisert

å beskrive, men vil ha store påvirkninger på hastighetsprofil. Ved saktegående trafikk i en køsituasjon er det typisk sjokkbølger som forplanter seg, som fører til start- og stoppfrekvenser for kjøretøyene. Evnen tunge kjøretøy har til å akselerere hurtig fra lav hastighet i motbakker er svært begrenset.

10.5.4 Intelligente hastighetsprofil

Kjøretøyenes kruskontroll blir stadig mer intelligent. Nye kjøretekniske tiltak blir implementert for å spare drivstoff, som beskrevet i kapittel 9.3. Vogntogforsøkene med Scania og Volvo ble gjort ved kruskjøring på økonomiprogrammene, så potensielle effekter av denne teknologien kan ha blitt inkludert i hastighetsmodellen. Det er likevel usikkerhet rundt effektene og grad av målbarhet til denne teknologien inn mot hastighetsmodellen i motbakker, med unntak av en girstrategi med få girskift.

10.6 Tilpassede girskift og motordata

Basert på motordata fra forsøksturene er det mulig å implementere hvert enkelt girskift manuelt. Dette er blitt gjort for Scania R580 i to av stigningene som ble kjørt under forsøksturene, på bakgrunn av reelt turtall før hvert enkelt girskift. Resultatet er presentert i figur 10.14 og 10.15, hvor oransje graf er hastighetsberegning med ny hastighetsmodell inkludert tilpassede girskift. Den røde og grønne grafen er henholdsvis ny hastighetsmodell uten tilpassede girskift og hastighetsprofil fra forsøksturene, og er like som i kapittel 10.4.5. Tilgjengelig motoreffekt er også tilpasset etter registrert motordata over stigningene.

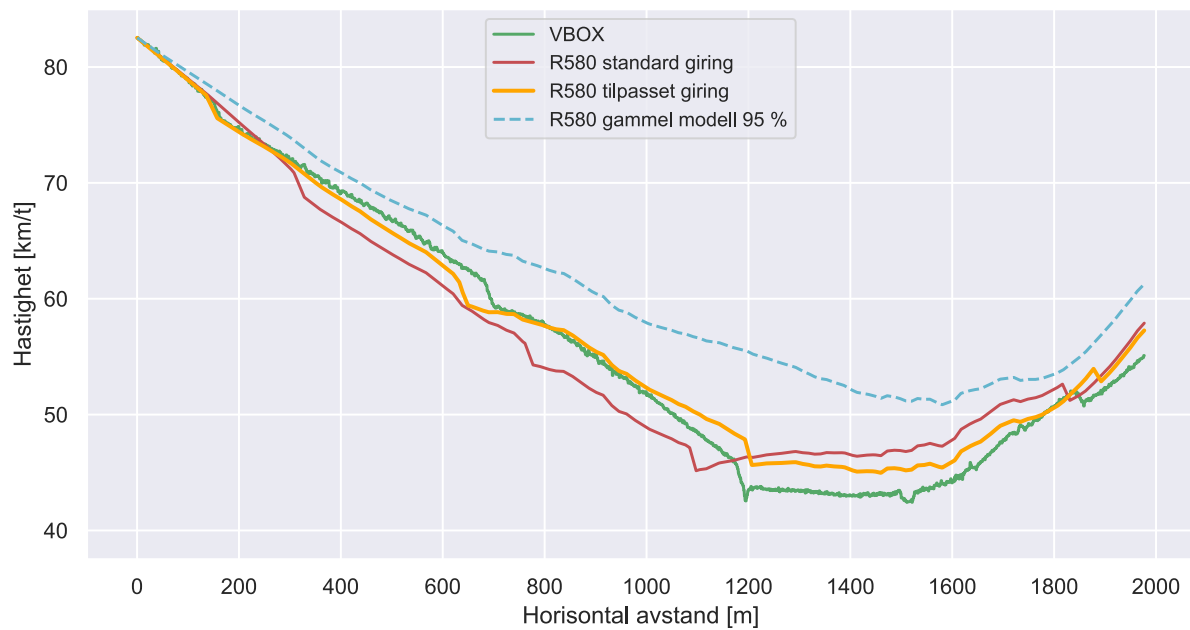
Under forsøksturen i Okstadbakken vist i figur 10.14 ble 98 % av den tilgjengelige motoreffekten utnyttet, som modellen er blitt justert for. Modellen girer ned fra 12. til 11. og fra 11. til 10. ved 1094 omdreininger. Girskift fra 10. til 9. skjer ved 1015 omdreininger. Fra 9. til 10. girer modellen ved 1435 omdreininger. Tallene er hentet direkte fra motordataen under forsøksturen. Formen på hastighetsprofilen fra modellen med tilpasset giring sammenfaller bedre med reelt hastighetsprofil fra forsøksturen, sammenlignet med modellen med standard giring. Dette gjenspeiles også i tilhørende GEH-verdier for modellen med tilpassede girskift og motordata i tabell 10.8, sammenlignet med standard giring.

For stigningen ved Vuddu vist i figur 10.15 ble mellom 92 og 99 % av motoreffekten utnyttet med en snittverdi på 96 %. Modellen med tilpasset giring gjennomfører kun ett girskift, direkte fra 11. til 9. gir ved 1032 omdreininger. Videre utvikler modellens hastighetsprofil seg nært identisk med reelt hastighetsprofil fra forsøksturen.

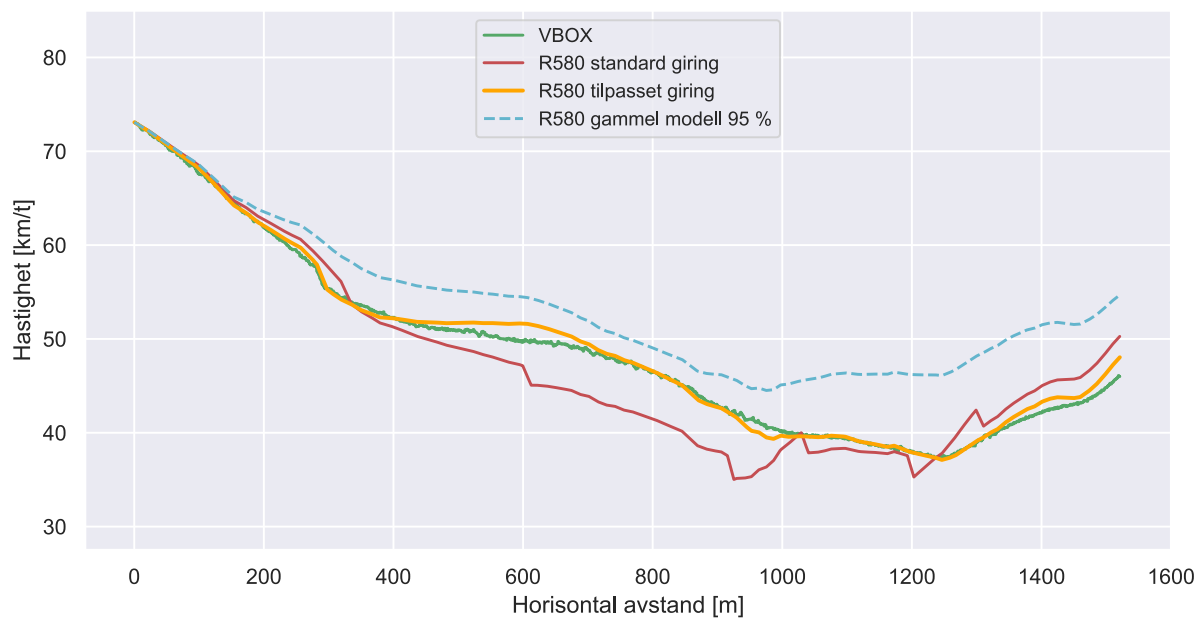
Disse eksemplene er ment for å vise potensialet modellen har for presis hastighetsmodellering, men utfordringene rundt “timing” av girskift må fortsatt løses. Tidspunkt for girskift vil i stor grad påvirke hastighetsprofilen til kjøretøyet og vil være individuelt for kjøretøy med ulike spesifikasjoner, last, inngangshastighet og ytre påvirkninger.

Stigning	GEH-verdi tilp. giring	GEH-verdi std. giring
Okstad	1,82	3,32
Vuddu	0,77	3,79

Tabell 10.8: GEH-verdier for hastighetsmodell med tilpassede girskift og motordata



Figur 10.14: Hastighetsmodell med tilpasset giring, Okstad, Scania R580



Figur 10.15: Hastighetsmodell med tilpasset giring, Vuddu, Scania R580

10.7 Endring av utnyttelsesgrad i eksisterende modell

Gjennom arbeidet med oppgaven er det en gjentakende trend at 95 % utnyttelsesgrad av maksimal tilgjengelig motoreffekt overestimerer hastigheten til tunge kjøretøy i motbakker. En justering av utnyttelsesgraden vil i teorien kunne gi et bedre estimat. Da kjøretøyene i hovedsak blir kjørt i et økonomiprogram for effektiv bruk av energi, vil det være naturlig å se på den tilgjengelige effekten ved turtallet der drivstofforbruket er lavest. I figur 10.6 vil dette punktet være på midten av plataået i momentgrafen. Dette kommer også frem av figur 8.11, hvor antall gram drivstoff per mengde omgjort energi (g/kWh) for Mercedes-Benz Arocs er oppgitt. Utnyttelsen av drivstoff er høyest ved omtrent 1200 omdreininger, som også er på midten av momentplataået.

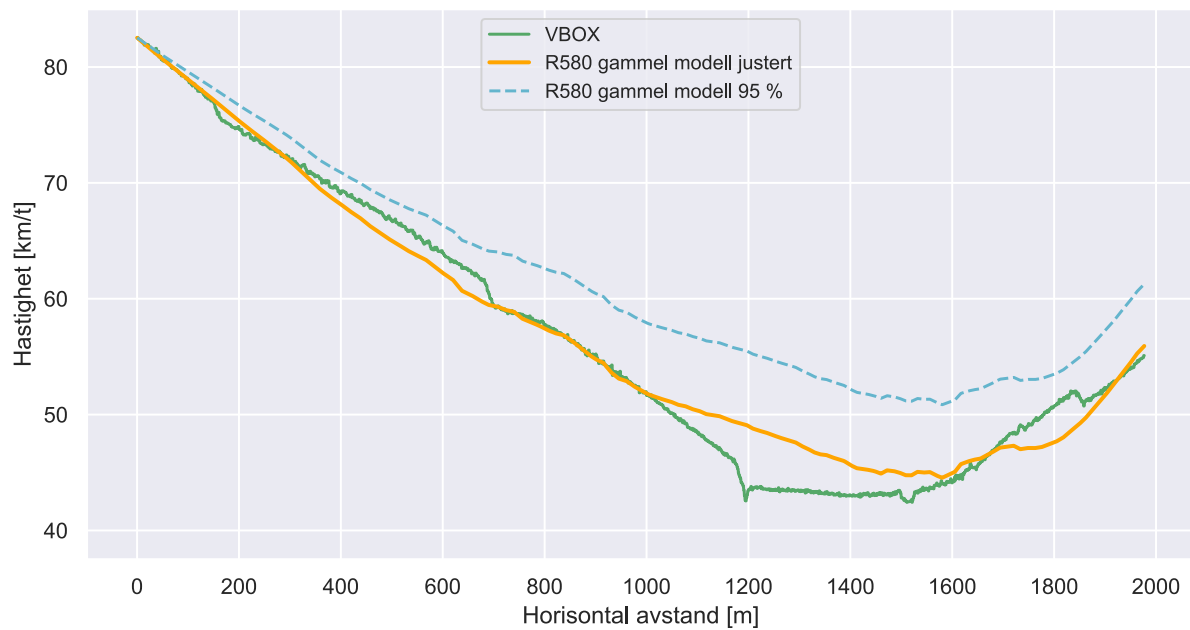
For å se om en justering av utnyttet effekt kan gi en bedre tilnærming er det valgt å se på de samme stigningene som i kapittel 10.6 med Scania R580. På midten av plataået for momentgrafen er turtallet 1125 omdreininger og motoreffekten vil være 356,5 kW som tilsvarer 83,9 % av maksimal effekt. Fra modelleringen med tilpasset giring i kapittel 10.6 er det beregnet gjennomsnittlig utnyttet effekt over hastighetsprofilene. Den gjennomsnittlige effekten er beregnet til henholdsvis 84,4 % og 80,5 % for stigningene ved Okstad og Vuddu. Dette indikerer at en utnyttelsesgrad på 83,9 % er et fornuftig estimat.

I figur 10.16 og 10.17 er eksisterende modell med 95 % utnyttelsesgrad av maksimal tilgjengelig motoreffekt (blå graf) satt opp mot eksisterende modell med 83,9 % utnyttelsesgrad (oransje graf), samt reelt hastighetsprofil fra forsøksturen med Scania R580 (grønn graf). Både visuelt og av GEH-verdier i tabell 10.9 er det tydelig at en reduksjon i utnyttelsesgraden til eksisterende modell gir langt bedre tilpassede hastighetsprofil sammenlignet med forsøksturene.

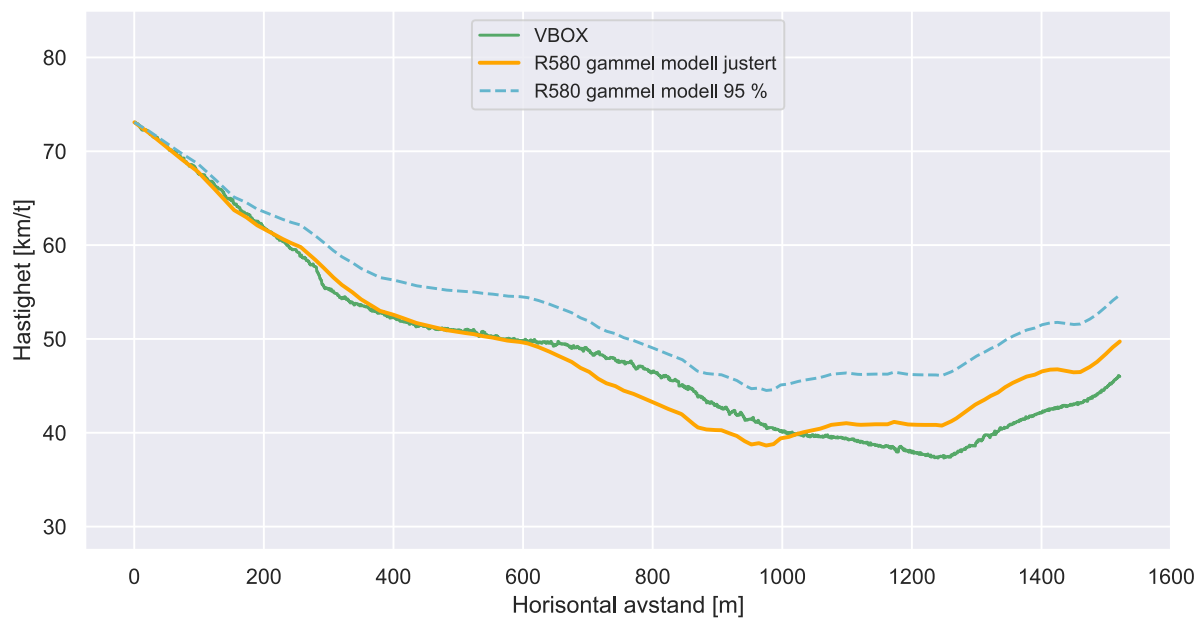
Det understrekes at dette resultatet er basert på kun to modellerte stigninger med ett spesifikt kjøretøy. Det kan ikke konkluderes direkte fra resultatet, men en slik reduksjon i utnyttelsesgrad fremstår logisk. Den tilgjengelige effekten er til en hver tid direkte avhengig av turtallet. Valgt kjøreprogram og gradienten til stigninger vil påvirke hvilket turtallsområde kjøretøyet opererer i for å oppnå akseptabel hastighet, samtidig som energien fra drivstoffet utnyttes mest mulig effektivt.

Stigning	GEH-verdi 83,9 %	GEH-verdi 95 %
Okstad	2,18	7,86
Vuddu	2,90	8,21

Tabell 10.9: GEH-verdier for hastighetsmodell med endret utnyttelsesgrad



Figur 10.16: Hastighetsmodell med justert utnyttelsesgrad, Okstad, Scania R580



Figur 10.17: Hastighetsmodell med justert utnyttelsesgrad, Vuddu, Scania R580

11 Oppsummering

Formålet med masteroppgaven har vært å videreutvikle hastighetsmodellen til Børnes og Aakre (2011) for tunge kjøretøy i motbakke, basert på erfaringsdata fra feltforsøk. På bakgrunn av formålet ble det definert to forskningsspørsmål som masteroppgaven skulle gi svar på:

1. Hvordan kan loggerutstyr benyttes for å innhente motordata fra tunge kjøretøy?
2. Hvordan kan motordata anvendes for å videreutvikle en hastighetsmodell for tunge kjøretøy i stigning?

Ulike tilnærminger til hvordan motordatalogging kunne gjennomføres ble kartlagt, med fokus på kjøretøyenes OBD-port og flåtestyringssystem (FMS). En kjent utfordring i forbindelse med CAN-tilkobling for motordatalogging i tunge kjøretøy er ulike praksiser for datatilgjengelighet mellom fabrikanter. I lys av denne utfordringen gikk prosjektet til anskaffelse av motorloggerutstyr med flere ulike tilkoblingsmuligheter. Eksternt utstyr for posisjonslogging ble benyttet og slapp unna denne problematikken.

Motordatalogging ble forsøkt gjennom OBD-porten på to eldre lastebiler ved NTNU, uten hell. Det ble i etterkant av dette gjennomført loggerforsøk på tre ulike lastebilfabrikanter. For Scania, som eneste fabrikant, var det tilgjengelig CAN-tilkobling direkte gjennom OBD-porten. For Mercedes-Benz ble motorloggerutstyr klipset over ledningene med CAN-data i forbindelse med kjøretøyets FMS-system. For Volvo ble løsningen å filme dashbordet for å registrere motordata, da det ikke var mulighet til å utforske metoden som fungerte for Mercedes-Benz.

Innsamlet data fra forsøksturene med de tre fabrikantene ble brukt til å videreutvikle eksisterende hastighetsmodell for tunge kjøretøy i motbakke. Resultatet er en modellspesifikk hastighetsmodell som tar hensyn til variabel motoreffekt og girskift. Motbakkefremstilling og rullestand i modellen er også videreutviklet for å stemme bedre overens med virkeligheten.

Den nye modellspesifikke hastighetsmodellen beregnet hastighetsprofil for motbakker som ble kjørt under forsøksturene med kjøretøy fra de ulike fabrikantene. Resultatet ble så satt opp mot faktisk registrert hastighet og beregnet hastighet ved bruk av den gamle modellen til Børnes og Aakre. Det er tydelig at den videreutviklede modellspesifikke hastighetsmodellen beregner hastighetsprofil som stemmer godt overens med faktisk registrert hastighet. Den gamle modellen til Børnes og Aakre er konsekvent optimistisk, med høyere hastighetsestimer enn hva som ble erfart under forsøksturene.

11.1 Videre arbeid

11.1.1 Motordatalogging

Prosjektet har presentert utstyr og ulike metoder for å logge motordata. Enkelte av disse metodene ble gjennomført i praksis under reelle loggerforsøk, med gode resultater. I prosjektet gikk det med mye arbeid til å forhøre seg med ulike lastebilfabrikanter om datagang og datatilgjengelighet i kjøretøyene, samt utprøving av ulike metoder for motordatalogging. En form for kartlegging av ulike lastebilfabrikanter med datagang og datatilgjengelighet vil kunne spare fremtidige arbeider som inkluderer motordatalogging fra tunge kjøretøy mye tid og ressurser.

11.1.2 Hastighetsmodell

Det er flere ulike områder som kan jobbes med for å videreutvikle den nye modellspesifikke hastighetsmodellen prosjektet har laget. Det er fremdeles interessant å betrakte tunge kjøretøy fra flere fabrikanter utover de dette prosjektet har gjort, for å se på loggermuligheter, operasjonelt turtallområde, dreiemoment- og effektkurver og teknologi for økonomisk kjøring.

Kartlegging av hastighetsprofiler med ulike vektkombinasjoner i de unike lastebilmodellene vil være interessant for å validere om det grunnleggende kinematiske oppsettet i hastighetsmodellen med kraftberegninger er innenfor rimelighetens grenser, eller trenger kalibrering.

Et annet interessant aspekt er å betrakte flere lastebiler av samme modell, men med ulik motorstørrelse. På den måten kan den modellspesifikke hastighetsmodellens skalerbarhet utforskes, med sentrale spørsmål rundt legitimiteten ved å skalere hastighetsmodellen direkte med ulike motorstørrelser.

Ved utforsking av flere tunge kjøretøy fra ulike fabrikanter med ulike vektkombinasjoner og motorstørrelser, kan et videre arbeidsområde være å vurdere dimensjonerende kjøretøy. Dette har ikke vært et fokusområde i prosjektet, men en konsekvens av en stadig fornyet tungtransportflåte er potensielt at reelt representativt kjøretøy beveger seg vekk fra hva Statens vegvesen har fastsatt som dimensjonerende.

Prosjektet erfarte gjennom loggerforsøkene at girvalgene som gjøres ved kjøring på kruskontroll nødvendigvis ikke er konsekvente i sammenheng med turtall. Stigningsgrad, vekt og stigningens utforming er blant faktorene som spiller inn på girstrategi og girvalg. Her er det muligheter for å gjøre girfaktoren i hastighetsmodellen avhengig av flere relevante parametere enn kun turtall, for å “time” girskift bedre i forhold til virkeligheten.

Teknologi for økonomisk kjøring som er drøftet i kapittel 9.3 har også innvirkning på girstrategi og oppnådd hastighetsprofil i stigninger. En mer dyptgående analyse av hvordan denne teknologien i nyere kjøretøy fungerer, samt påvirkning på girstrategi, hastighet og kjørestil vil være nyttig. Om ny teknologi fører til konsekvente kjøretekniske handlinger for å spare drivstoff, er det viktig å få kartlagt disse inn mot hastighetsmodeller og dimensjonering.

Referanser

- 3Rtablet (2021). *FMS-interface standard connector description*. (Hentet 04.02.2020). URL: <https://www.3rtablet.com/support/faq/fms-interface-standard-connector-description.html>.
- Ai, Jun mfl. (2011). «Assessment of rolling resistance models in discrete element simulations». I: *Powder Technology* 206.3, s. 269–282.
- Archilla, Adrián Ricardo og Aristides Osvaldo Fernandez De Cieza (1996). «Truck performance on Argentinean highways». I: *Transportation Research Record* 1555.1, s. 114–123.
- Arellano, Daniela, Tomás Echaveguren og Sergio Vargas-Tejeda (2015). «A model of truck speed profiles on short upward slopes». I: *Proceedings of the Institution of Civil Engineers-Transport*. Bd. 168. 5. Thomas Telford Ltd, s. 475–483.
- Bester, Christo J (2000). «Truck speed profiles». I: *Transportation Research Record* 1701.1, s. 111–115.
- Børnes, Vilhelm og Arvid Aakre (2011). «Description, validation and use of a model to estimate speed profile of heavy vehicles in grades». I: *Procedia-social and behavioral sciences* 16, s. 409–418.
- Copperhill Technologies (2020). *SAE J1939/13 Off-Board Diagnostic Connector*. (Hentet 25.01.2020). URL: <https://copperhilltech.com/blog/sae-j193913-offboard-diagnostic-connector-deutsch-hd1091939/>.
- Copperhill technologies (2020). *A Brief Introduction to the SAE J1939 Protocol*. (Hentet 10.11.2020). URL: <https://copperhilltech.com/a-brief-introduction-to-the-sae-j1939-protocol/>.
- CSS Electronics (2020a). *CANedge1 documentation*. (Hentet 21.11.2020). URL: <https://canlogger.csselectronics.com/canedge-docs/ce1/introduction.html>.
- (2020b). *CANedge1: 2x CAN Bus Data Logger (SD + RTC)*. (Hentet 17.12.2020). URL: <https://www.csselectronics.com/screen/product/can-logger-sd-canedge1/language/en>.
 - (2020c). *DB9-Deutsch 9-Pin Cable (Type 2 J1939, CAT)*. (Hentet 17.12.2020). URL: <https://www.csselectronics.com/screen/product/can-bus-logger-j1939-adaptor>.
 - (2020d). *J1939 DBC File - PGN & SPN Data*. (Hentet 21.11.2020). URL: <https://www.csselectronics.com/screen/product/j1939-dbc-file-pgn-spn>.
 - (2021a). *CANCrocodile - Contactless CAN Bus Reader*. (Hentet 09.02.2020). URL: <https://www.csselectronics.com/screen/product/cancrocodile-contactless-can-adapter#void>.
 - (2021b). *OBD2 Explained - A Simple Intro (2021)*. (Hentet 21.1.2021). URL: <https://www.csselectronics.com/screen/page/simple-intro-obd2-explained/language/en>.
 - (2021c). *OBD2-to-DB9 Adapter Cable*. (Hentet 4.3.2021). URL: <https://www.csselectronics.com/screen/product/can-bus-logger-obd-adapter>.
- Gillespie, Thomas D (1992). *Fundamentals of vehicle dynamics*. Bd. 400. Society of automotive engineers Warrendale, PA.
- Hammarström, Ulf mfl. (2012). *Rolling resistance model, fuel consumption model and the traffic energy saving potential from changed road surface conditions*. Statens väg-och transportforskningsinstitut.
- Hrisca, Daniel (2018). *GUI*. (Hentet 14.04.2021). URL: <https://asammdf.readthedocs.io/en/latest/gui.html>.
- LaClair, TJ (2005). «Rolling resistance». I: *The pneumatic tire*, s. 475–532.
- Lan, Chang-Jen og Monica Menendez (2003). «Truck speed profile models for critical length of grade». I: *Journal of Transportation Engineering* 129.4, s. 408–419.

- Mercedes-Benz (2021). *The Arocs - Engine performance data*. (Hentet 25.04.2021). URL: https://www.mercedes-benz-trucks.com/en_GB/models/arocs/technical-data/engine-performance-data.html.
- Nesdal, Andreas (2019). «Modellering av fartsprofil til tunge køyretøy i stigning». Masteroppg. NTNU.
- Nygaard, Simen W. og Jakob Kogstad (2020). *Modellering av tunge kjøretøy i stigninger*. Prosjektoppg. NTNU. Upublisert.
- Opstad, Erlend (2015). «Behov for forbikjøringsfelt i stigning på E39 Ørskogfjellet». Masteroppg. NTNU.
- Prasad, BVP, Jing-Jou Tang og Sheng-Jhu Luo (2019). «Design and Implementation of SAE J1939 Vehicle Diagnostics System». I: *2019 IEEE International Conference on Computation, Communication and Engineering (ICCCE)*. IEEE, s. 71–74.
- Racelogic (2020a). *VBOX Mini*. (Hentet 17.12.2020). URL: <https://www.vboxautomotive.co.uk/index.php/en/products/data-loggers/vbox-mini>.
- (2020b). *VBOX Vehicle Testing Products*. (Hentet 17.12.2020). URL: <https://www.vboxautomotive.co.uk/index.php/en/products>.
- Rakha, Hesham og Ivana Lucic (2002). «Variable power vehicle dynamics model for estimating truck accelerations». I: *Journal of transportation engineering* 128.5, s. 412–419.
- Rakha, Hesham, Ivana Lucic mfl. (2001). «Vehicle dynamics model for predicting maximum truck acceleration levels». I: *Journal of transportation engineering* 127.5, s. 418–425.
- Robert Bosch GmbH (2007). *Automotive Handbook 7th edition*. Robert Bosch GmbH.
- Sawant, Pooja Rajendra og Yashwant B Mane (2018). «Design and development of on-board diagnostic (OBD) device for cars». I: *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*. IEEE, s. 1–4.
- Scania (2018). *How to use the truck's kinetic energy to save fuel*. (Hentet 13.04.2021). URL: <https://www.scania.com/group/en/home/newsroom/news/2018/how-to-use-the-trucks-kinetic-energy-to-save-fuel.html>.
- Schramm, Dieter, Manfred Hiller og Roberto Bardini (2014). «Vehicle dynamics». I: *Modeling and Simulation*. Springer, s. 151.
- Vegdirektoratet, Statens vegvesen (2019a). *Håndbok N100 Veg-og gateutforming*.
- (2019b). *Håndbok V120 Premisser for geometrisk utforming av veger*. Tekn. rapp.
- Volvo (2021). *Drivlinje*. (Hentet 27.04.2021). URL: https://www.volvotrucks.no/content/dam/volvo/volvo-trucks/masters/euro-6/pdf/trucks/volvo-fh-series/driveline/Volvo-FH-Serien_Spesifikasjoner_Drivlinje_NO.pdf.
- Xin, Cai mfl. (2020). «Optimization on design parameters of road longitudinal slope based on truck dynamics». I: *Green, Smart and Connected Transportation Systems*. Springer, s. 1325–1343.

Vedlegg A Innlesning av CSV-filer

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\CSV_File_Reader.py

```
1
2 import pandas as pd
3 import numpy as np
4 import matplotlib.pyplot as plt
5 import math
6 import seaborn as sns
7 sns.set_theme()
8 save_results_to = '/Users/simen/Desktop/Plot/'
9 # t = np.arange(0.1,182.4,0.1).tolist()
10 # VBOX_Speed = ['Longitude']
11 # CSS_Speed = ['S1_PID_0D_VehicleSpeed']
12 # data_VBOX = pd.read_csv(r'C:\Users\simen\Desktop\VBOX_062.csv',usecols=VBOX_Speed)
13 # data_CSS = pd.read_csv(r'C:\Users\simen\Desktop\second time change test export.
    ChannelGroup_7_0BD2_0x7E8.csv',usecols=CSS_Speed)
14 #
15 # data_VBOX = pd.read_csv(r'C:\Users\simen\Desktop\15 mars\VBOX_069.csv',usecols=VBOX_Speed)
16 # data_CSS = pd.read_csv(r'C:\Users\simen\Desktop\second time change test export.
    ChannelGroup_7_0BD2_0x7E8.csv',usecols=CSS_Speed)
17 #
18 # list_VBOX = data_VBOX['Longitude'].values.tolist()
19 # list_CSS = data_CSS['S1_PID_0D_VehicleSpeed'].values.tolist()
20 # print(list_VBOX)
21 #
22 # Max = max(list_VBOX + list_CSS)
23 #
24 # diff = []
25 #
26 def round_half_up(n, decimals=0):
27     multiplier = 10 ** decimals
28     return math.floor(n*multiplier + 0.5) / multiplier
29 #
30 # zip_list = zip(list_VBOX,list_CSS)
31 # for list_VBOX_i, list_CSS_i in zip_list:
32 #     diff.append(round_half_up(abs(list_VBOX_i-list_CSS_i), 2))
33 #
34 #
35 # print(Max)
36 # print(diff)
37 #
38 # fig, ax1 = plt.subplots()
39 #
40 # color = 'tab:red'
41 # ax1.set_xlabel('time (s)')
42 # ax1.set_ylabel('Speed VBOX', color=color)
43 # ax1.plot(t, data_VBOX, color=color)
44 # ax1.tick_params(axis='y', labelcolor=color)
45 #
46 # ax2 = ax1.twinx() # instantiate a second axes that shares the same x-axis
47 # ax1.set_ylim(0,Max + 5)
48 # ax2.set_ylim(0,Max + 5)
49 # color = 'tab:blue'
50 # ax2.set_ylabel('Speed CSS', color=color) # we already handled the x-label with ax1
51 # ax2.plot(t, data_CSS, color=color)
52 # ax2.tick_params(axis='y', labelcolor=color)
53 #
54 # fig.tight_layout() # otherwise the right y-label is slightly clipped
55 # plt.show()
56
57 # --- Filnavn -----
58 # filename = "\Tillerbakken.csv"
59 # filename = "\Scania_0kstadbakken_100.csv"
60 filename = "\Vuddudalen_Scania_100.csv"
```


File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\CSV_File_Reader.py

```
61 # filename = "\Høllanbakken_2.csv"
62 # filename = "\Volvo_0kstadbakken_2.csv"
63 # filename = "\Sveberg_Scania_100.csv"
64 # filename = "\0kstadbakken_Mercedes_Arocs_Full.csv"
65 # filename = "\Moholtbakken_Mercedes_full.csv"
66 # filename = "\Arocs_Klett_til_Moholt.csv"
67 # filename = "\Klettbakken_Arocs.csv"
68 # filename = "\Bakkeprofil_Tiller_2.csv"
69
70 BASE_PATH = r"C:\Users\simen\Desktop"
71 READ_FILE = BASE_PATH + filename
72
73 VBOX_Speed = ['Velocity']
74 data_VBOX = pd.read_csv(READ_FILE, usecols=VBOX_Speed, index_col=False)
75 speed_VBOX = data_VBOX['Velocity'].values.tolist()
76
77 VBOX_distance = ['Distance']
78 VBOX_height = ['Height']
79 distance_VBOX = pd.read_csv(READ_FILE, usecols=VBOX_distance, index_col=False)
80 height_VBOX = pd.read_csv(READ_FILE, usecols=VBOX_height, index_col=False)
81
82 # fig, ax1 = plt.subplots()
83 #
84 # color = 'tab:red'
85 # ax1.set_xlabel('Distance [m]')
86 # ax1.set_ylabel('Height [m]', color=color)
87 # ax1.plot(distance_VBOX, height_VBOX, color=color)
88 # ax1.tick_params(axis='y', labelcolor=color)
89 #
90 # fig.tight_layout() # otherwise the right y-label is slightly clipped
91 # plt.show()
92
93 list_distance = distance_VBOX['Distance'].values.tolist()
94 list_distance = [x-list_distance[0] for x in list_distance]
95 list_height = height_VBOX['Height'].values.tolist()
96
97 # print(list_height)
98
99 acchor = 0
100 deltah = []
101 deltax = []
102 stigning = []
103 Acchor = []
104 i = 0
105
106 for height in range(len(list_height)-1):
107     dh = list_height[i+1] - list_height[i]
108     deltah.append(dh)
109     dx = list_distance[i+1] - list_distance[i]
110     deltax.append(dx)
111     acchor = dx + acchor
112     Acchor.append(acchor)
113     if dx == 0:
114         s = 0
115     else:
116         s = dh/dx * 100
117     if abs(s) > 15:
118
119         s = 0
120
121     stigning.append(s)
122     i = i+1
```

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\CSV_File_Reader.py

```
123
124 # Hastighet VBOX -----
125 fig, ax1 = plt.subplots()
126
127 color = 'tab:red'
128 ax1.set_xlabel('Horisontal avstand [m]')
129 ax1.set_ylabel('Hastighet [km/t]', color=color)
130 ax1.plot(Acchor, speed_VBOX[1:], color=color)
131 ax1.tick_params(axis='y', labelcolor=color)
132
133 fig.tight_layout() # otherwise the right y-label is slightly clipped
134 plt.show()
135
136 # Stigning -----
137 fig, ax1 = plt.subplots()
138
139 ax1.set_xlabel('Horisontal avstand [m]')
140 ax1.set_ylabel('Stigning [%]')
141 ax1.plot(Acchor, stigning, 'b')
142 ax1.tick_params(axis='y')
143 ax1.set_title('VBOX stigning alle loggepunkter')
144
145 fig.tight_layout() # otherwise the right y-label is slightly clipped
146 plt.savefig(save_results_to + 'VBOX_stigning_alle_loggepunkter.pdf')
147 plt.show()
148
149 # Gjennomsnitt stigning 10 -----
150 stigning10array = np.mean(np.array(stigning).reshape(-1,10), axis=1)
151 stigning10 = stigning10array.tolist()
152 Acchor10 = Acchor[9::10]
153
154 fig, ax1 = plt.subplots()
155
156 ax1.set_xlabel('Distance [m]')
157 ax1.set_ylabel('Stigning [%]')
158 ax1.plot(Acchor10, stigning10, 'b')
159 ax1.tick_params(axis='y')
160 fig.tight_layout() # otherwise the right y-label is slightly clipped
161 plt.savefig(save_results_to + 'Stigning_10_snitt.pdf')
162 plt.show()
163
164 # stigning20array = np.mean(np.array(stigning).reshape(-1,20), axis=1)
165 # stigning20 = stigning20array.tolist()
166 # Acchor20 = Acchor[19::20]
167
168 # fig, ax1 = plt.subplots()
169 # ax1.set_xlabel('Distance [m]')
170 # ax1.set_ylabel('Stigning [%]')
171 # ax1.plot(Acchor20, stigning20, 'g')
172 # ax1.tick_params(axis='y')
173 # fig.tight_layout() # otherwise the right y-label is slightly clipped
174 # plt.savefig(save_results_to + 'Stigning_20_snitt.pdf')
175 # plt.show()
176 #
177 # stigning100array = np.mean(np.array(stigning).reshape(-1,100), axis=1)
178 # stigning100 = stigning100array.tolist()
179 # Acchor100 = Acchor[99::100]
180 #
181 # fig, ax1 = plt.subplots()
182 #
183 # color = 'tab:blue'
184 # ax1.set_xlabel('Distance [m]')
```

```
File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\CSV_File_Reader.py
185 # ax1.set_ylabel('Stigning100 dh/dx', color=color)
186 # ax1.plot(Acchor100, stigning100, color=color)
187 # ax1.tick_params(axis='y', labelcolor=color)
188 #
189 # fig.tight_layout() # otherwise the right y-label is slightly clipped
190 # plt.show()
191
192 # print(Acchor[math.floor(10.4)])
```

Vedlegg B Stigningsberegning med NVDB-data

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\NVDB_posisjon.py

```
1 import pandas as pd
2 import math
3 import matplotlib.pyplot as plt
4
5 data = pd.read_csv(r'C:\Users\simen\Desktop\Radata_stigning\Tillerbakken_sammenligning.txt',
6 header=None)
7 liste = data.values.tolist()
8 liste2 = liste[0]
9 liste2.reverse() #snur listen i de tilfellene man går mot metreringslinjen
10
11 #--- Endres til z y x når man går mot metreringslinjen ---
12 z = liste2[0::3]
13 y = liste2[1::3]
14 x = liste2[2::3]
15
16 Hor = []
17 acchor = 0
18 Acchor = []
19 S = []
20 c = 0
21 c2 = 0
22
23 for posisjon in range(len(x) - 1):
24     hor = math.sqrt((x[c + 1] - x[c])**2 + (y[c + 1] - y[c])**2)
25     Hor.append(hor)
26     acchor = hor + acchor
27     Acchor.append(acchor)
28     s = (z[c + 1] - z[c]) / hor * 100
29     S.append(s)
30
31     c = c + 1
32     # print(c)
33     # print(x[c])
34     # print(y[c])
35
36 Q = [0,]
37 L = Q + Acchor
38
39 fig, ax1 = plt.subplots()
40
41 color = 'tab:green'
42 ax1.set_xlabel('Horisontal avstand [m]')
43 ax1.set_ylabel('Høyde [m]', color=color)
44 ax1.plot(L, z, color=color)
45 ax1.tick_params(axis='y', labelcolor=color)
46
47 fig.tight_layout() # otherwise the right y-label is slightly clipped
48 plt.show()
49
50
51 fig, ax1 = plt.subplots()
52
53 color = 'tab:red'
54 ax1.set_xlabel('Horisontal avstand [m]')
55 ax1.set_ylabel('Stigning NVDB [%]')
56 ax1.plot(Acchor, S, 'r')
57 ax1.tick_params(axis='y')
58
59 fig.tight_layout() # otherwise the right y-label is slightly clipped
60
61 plt.show()
```

Vedlegg C Stigningssammenligning VBOX-NVDB

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\Stigningssammenligning.py

```
1 import matplotlib.pyplot as plt
2 plt.rcParams.update({'font.size': 16})
3
4
5 from Hastighetsmodell import NVDB_posisjon
6 from Hastighetsmodell import CSV_File_Reader
7
8 Z_NVDB = NVDB_posisjon.z
9 X_NVDB = NVDB_posisjon.L
10
11 Z_VBOX = CSV_File_Reader.list_height
12 X_VBOX = CSV_File_Reader.list_distance
13
14 Z_diff = [x - (Z_VBOX[0] - Z_NVDB[0]) for x in Z_VBOX]
15 null_Z_VBOX = [(x - Z_VBOX[0]) for x in Z_VBOX]
16 null_Z_NVDB = [(x - Z_NVDB[0]) for x in Z_NVDB]
17
18 save_results_to = '/Users/simen/Desktop/Plot/'
19
20 fig1, ax1 = plt.subplots()
21
22 ax1.set_xlabel('Horisontal avstand [m]', fontsize=16)
23 ax1.set_ylabel('Høydemeter [moh]', fontsize=16)
24 plt.plot(X_NVDB, Z_NVDB, 'r', label='NVDB') # plotting t, a separately
25 plt.plot(X_VBOX, Z_VBOX, 'b', label='VBOX') # plotting t, b separately
26 ax1.tick_params(labelsize=14)
27 plt.legend(fontsize=16)
28 plt.savefig(save_results_to + 'NVDBvsVBOX.pdf', bbox_inches='tight')
29 plt.show()
30
31
32 fig2, ax2 = plt.subplots()
33
34 ax2.set_xlabel('Horisontal avstand [m]', fontsize=16)
35 ax2.set_ylabel('Vertikal avstand [m]', fontsize=16)
36 plt.plot(X_NVDB, null_Z_NVDB, 'r', label='NVDB') # plotting t, a separately
37 plt.plot(X_VBOX, null_Z_VBOX, 'b', label='VBOX') # plotting t, b separately
38 ax2.tick_params(labelsize=14)
39 plt.legend(fontsize=16)
40 plt.savefig(save_results_to + 'NVDBvsVBOX_0.pdf', bbox_inches='tight')
41 plt.show()
42
43 fig3, ax3 = plt.subplots()
44
45 ax3.set_xlabel('Horisontal avstand [m]')
46 ax3.set_ylabel('Stigning [%]')
47 plt.plot(NVDB_posisjon.L[1:], NVDB_posisjon.S, 'r', label='NVDB') # plotting t, a separately
48 plt.plot(CSV_File_Reader.Achcor20, CSV_File_Reader.stigning20, 'b', label='VBOX') # plotting t
49 # b separately
50 plt.legend()
51 plt.savefig(save_results_to + 'Stigningssammenligning.pdf', bbox_inches='tight')
52 plt.show()
53
54 print('VBOX høyde start ', Z_VBOX[0])
55 print('NVDB høyde start ', Z_NVDB[0])
56
57 print('VBOX høyde slutt', Z_VBOX[600])
58 print('NVDB høyde slutt', Z_NVDB[32])
59
60 print('VBOX avstand slutt', X_VBOX[600])
61 print('NVDB avstand slutt', X_NVDB[32])
```

```
File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\Stigning_sammenligning.py  
62 print('VB0X avstand lik lengde', X_VB0X[584])
```

Vedlegg D Listegenerering med iri- og mpd-data

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\IRI_MPD.py

```
1 import pandas as pd
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import math
5
6 filename = "\IRI_MPD_TXT.txt"
7
8 BASE_PATH = r"C:\Users\simen\Desktop"
9 READ_FILE = BASE_PATH + filename
10
11 IRI = ['IRI']
12 data_IRI = pd.read_csv(READ_FILE, usecols=IRI)
13 IRI_list = data_IRI['IRI'].values.tolist()
14
15 MPD = ['MPD']
16 data_MPD = pd.read_csv(READ_FILE, usecols=MPD)
17 MPD_list = data_MPD['MPD'].values.tolist()
18 MPD_list = [x/100 for x in MPD_list]
19
20 p = np.linspace(20, 20*len(IRI_list), len(IRI_list))
21 P = p.tolist()
22
23 print(len(P))
24 print(P[71])
```

Vedlegg E Modellspesifikk hastighetsmodell

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\Modell.py

```
1 import pandas as pd
2 import math
3 import time
4 start = time.time()
5 import matplotlib.pyplot as plt
6 plt.rcParams.update({'font.size': 12})
7 import CSV_File_Reader
8 import IRI_MPD
9 # import NVDB_posisjon
10 import numpy as np
11 from statistics import mean
12 import seaborn as sns
13 sns.set_theme()
14 # sns.set_style('')
15 import matplotlib.ticker as ticker
16 save_results_to = '/Users/simen/Desktop/Plot/'
17 import math
18
19 # --- Generering av høydeprofil med NVDB-data -----
20 # data = pd.read_csv(r'C:\Users\simen\Desktop\Radata_stigning\Nardobakken3.txt', header=None)
21 # for i in range(0, len(liste)):
22 #     for j in range(0, len(liste[i])):
23 #         x = liste[i][j]
24 # print(data)
25 #
26 # liste = data.values.tolist()
27 # liste2 = liste[0]
28 # liste2.reverse() #snur listen i de tilfellene man går mot metrerings\ linjen
29 #
30 # --- Endres til z y x når man går mot metreringslinjen.
31 # z = liste2[0::3]
32 # y = liste2[1::3]
33 # x = liste2[2::3]
34 #
35 # Hor = []
36 # acchor = 0
37 # ll = []
38 # ss = []
39 # c = 0
40 # c2 = 0
41 #
42 # for posisjon in range(len(x) - 1):
43 #     hor = math.sqrt((x[c + 1] - x[c])**2 + (y[c + 1] - y[c])**2)
44 #     Hor.append(hor)
45 #     acchor = hor + acchor
46 #     ll.append(acchor)
47 #     s = (z[c + 1] - z[c]) / hor * 100
48 #     ss.append(s)
49 #
50 #     c = c + 1
51 #
52 # -----
53 # --- Sveberg ---
54 # def get_Gir_Sveberg(Gir):
55 #     if rpm < 800:
56 #         Gir = Gir - 1
57 #     # elif rpm < 1250 and Gir == 12 and 800 < x < 1000:
58 #     #     Gir = Gir - 1
59 #     elif rpm > 1550 and Gir == 9:
60 #         Gir = Gir + 1
61 #     elif rpm > 1450 and Gir == 10:
62 #         Gir = Gir + 1
```


File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\Modell.py

```
63 #     elif rpm > 1400 and Gir == 11:
64 #         Gir = Gir + 1
65 #     return Gir
66
67 # --- Vuddu ---
68
69 def get_Gir_Vuddu(Gir):
70     if rpm < 1032 and d == 0 and x < 1000:
71         Gir = Gir - 2
72     elif rpm > 1600 and Gir != 12 and d == 0:
73         Gir = Gir + 1
74     return Gir
75
76 # --- Okstad ---
77 def get_Gir_Okstad(Gir):
78     if rpm < 1094 and Gir == 12 and d == 0:
79         Gir = Gir - 1
80     elif rpm < 1094 and Gir == 11 and d == 0:
81         Gir = Gir - 1
82     elif rpm < 1015 and Gir == 10 and d == 0:
83         Gir = Gir - 1
84     elif rpm > 1435 and Gir != 12 and d == 0:
85         Gir = Gir + 1
86     return Gir
87
88 # --- Standard ---
89 def get_Gir_standard(Gir):
90     if rpm < 1000 and d == 0:
91         Gir = Gir - 1
92     elif rpm > 1400 and Gir != 12 and d == 0:
93         Gir = Gir + 1
94     return Gir
95
96 # --- Sveberg ---
97 def get_Gir_Sveberg(Gir):
98     if Gir == 10 and rpm < 980 and d == 0:
99         Gir = Gir - 1
100     elif Gir != 10 and rpm < 1000 and d == 0:
101         Gir = Gir - 1
102     elif Gir == 9 and rpm > 1330 and d == 0:
103         Gir = Gir + 1
104     elif Gir == 10 and rpm > 1200 and x > 600 and d == 0:
105         Gir = Gir + 1
106     return Gir
107
108 # --- Scania -----
109
110 def get_rpm_eff(Gir):
111     if Gir == 12:
112         rpm_eff = 14.1
113     elif Gir == 11:
114         rpm_eff = 17.8
115     elif Gir == 10:
116         rpm_eff = 21.2
117     elif Gir == 9:
118         rpm_eff = 26.6
119     elif Gir == 8:
120         rpm_eff = 35.0
121     elif Gir == 7:
122         rpm_eff = 44
123     elif Gir == 6:
124         rpm_eff = 53
```

```

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\Modell.py
125     elif Gir == 5:
126         rpm_eff = 67.5
127     elif Gir == 4:
128         rpm_eff = 82.5
129     return rpm_eff
130
131 def get_P_w(rpm):
132     if 800 <= rpm < 950:
133         P_w = ((rpm - 800) * 0.441 * 1000) + 235.3 * 1000
134     elif 950 <= rpm <= 1300:
135         P_w = ((rpm - 950) * 0.315 * 1000) + 301.5 * 1000
136     elif 1300 < rpm < 1900:
137         P_w = ((rpm - 1300) * 0.0245 * 1000) + 411.8 * 1000
138     return P_w
139
140 # --- Mercedes -----
141
142 # def get_Gir(Gir):
143 #     if rpm < 1200 and d == 0:
144 #         Gir = Gir - 1
145 #     elif rpm > 1600 and Gir != 12 and d == 0:
146 #         Gir = Gir + 1
147 #     return Gir
148 #
149 # def get_rpm_eff(Gir):
150 #     if Gir == 12:
151 #         rpm_eff = 16.3
152 #     elif Gir == 11:
153 #         rpm_eff = 21.1
154 #     elif Gir == 10:
155 #         rpm_eff = 26.2
156 #     elif Gir == 9:
157 #         rpm_eff = 33.95
158 #     elif Gir == 8:
159 #         rpm_eff = 43.4
160 #     return rpm_eff
161 #
162 # def get_P_w(rpm):
163 #     if 800 <= rpm < 1000:
164 #         P_w = ((rpm - 800) * 0.575 * 1000) + 175 * 1000
165 #     elif 1000 <= rpm < 1550:
166 #         P_w = ((rpm - 1000) * 0.245 * 1000) + 290 * 1000
167 #     elif 1550 <= rpm <= 1800:
168 #         P_w = 424.75 * 1000
169 #     return P_w
170 #
171 # --- Volvo -----
172
173 # def get_Gir(Gir):
174 #     if rpm < 1000 and d == 0:
175 #         Gir = Gir - 1
176 #     elif rpm > 1450 and Gir != 12 and d == 0:
177 #         Gir = Gir + 1
178 #     return Gir
179 #
180 # def get_rpm_eff(Gir):
181 #     if Gir == 12:
182 #         rpm_eff = 15.52
183 #     elif Gir == 11:
184 #         rpm_eff = 19.78
185 #     elif Gir == 10:
186 #         rpm_eff = 25.31

```

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\Modell.py

```
187 #     elif Gir == 9:
188 #         rpm_eff = 32.25
189 #     elif Gir == 8:
190 #         rpm_eff = 41.92
191 #     elif Gir == 7:
192 #         rpm_eff = 53.40
193 #     elif Gir == 6:
194 #         rpm_eff = 67.50
195 #     elif Gir == 5:
196 #         rpm_eff = 85.20
197 #     elif Gir == 4:
198 #         rpm_eff = 110.15
199 #     elif Gir == 3:
200 #         rpm_eff = 140.2
201 #     elif Gir == 2:
202 #         rpm_eff = 183
203 #     elif Gir == 1:
204 #         rpm_eff = 231
205 #     return rpm_eff
206 #
207 # def get_P_w(rpm):
208 #     if 600 <= rpm < 1000:
209 #         P_w = ((rpm - 600) * 0.469 * 1000) + 80.8 * 1000
210 #     elif 1000 <= rpm <= 1450:
211 #         P_w = ((rpm - 1000) * 0.286 * 1000) + 268.4 * 1000
212 #     elif 1450 < rpm < 1900:
213 #         P_w = 397 * 1000
214 #     return P_w
215 #
216 # -----
217 #
218 ll = CSV_File_Reader.Acchor10
219 ss = CSV_File_Reader.stigning10
220 #
221 # ll = NVDB_posisjon.Acchor
222 # ss = NVDB_posisjon.S
223 #
224 # ll = [400,800,1400,1800,2800]
225 # ss = [0,3.5,7,3.5,0]
226 #
227 # ll = [2000]
228 # ss = [0]
229 speed_VBOX = CSV_File_Reader.speed_VBOX
230 #
231 # -----
232 # Scania_530 = []
233 # Scania_580 =
234 # Scania_
235 #
236 # --- Alternative profiler -----
237 # ll = [1000,2000,3000,4000,5000]
238 # ll = [50,140,356,475,708,908,1023,1489,1763,1906]
239 #
240 # --- Klett ---
241 # ll = [600,245,651,475,255,1326,245,401,350]
242 # ll = [600,845,1496,1971,2226,3552,3797,4198,4548]
243 # ss = [0.8,1.7,1.6,1.2,3.6,5.9,3.3,1.7,2.0]
244 #
245 # --- Nesdal ---
246 # ll = [196, 285, 1068, 1279, 1379]
247 # ss = [2.0, 4.5, 5.5, 3.3, 1.0]
248 # --- Nesdal mod ---
```

```

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\Modell.py
249 # ll = [196, 285, 1068, 1279, 1379, 1500, 1700, 2000, 2200, 2700]
250 # ss = [2.0, 4.5, 5.5, 3.3, 1.0, 0.0, -1.0, -3.0, -1.0, 0.0]
251 #
252 # --- Hommelvik ---
253 # ll = [1969, 2974]
254 # ss = [3.1, 4.6]
255 #
256
257 # --- Basisparametere -----
258 startgir = 11
259 # --- Vekt ---
260 m = 50000
261 # --- Rullemotstand Alt. 1 ---
262 k_r = 0.001
263 k_s = 1 / 4470
264 # --- Rullemotstand Alt. 2 ---
265 iri = 1.2
266 mpd = 1.3
267 # --- Luftmotstand ---
268 A = 8
269 cw = 0.6
270 rho = 1.2
271 # --- Vindhastighet ---
272 v0 = 0
273 # --- Tyngdekraft ---
274 g = 10
275 # --- Grenseverdier ---
276 v_max = 90
277 a_max = 3.0
278 # --- Starthastighet ---
279 v_start = speed_VBOX[0]
280 # v_start = 60
281 # --- Startpunkt ---
282 x1 = 0
283 h1 = 0
284 # --- Tidsteg ---
285 deltaT = 0.01
286 # --- Effekt gammel modell [kW]---
287 eff_gammel = 425
288
289 # Oppsett = [1,1,1,1,1]
290 # Oppsett = [0.7,1,1.138,1.326, 1]
291 Oppsett = [0.96,1,1,1,1]
292 # --- Lister ---
293
294 T = [[] for _ in range(len(Oppsett))]
295 RPM = [[] for _ in range(len(Oppsett))]
296 GIR = [[] for _ in range(len(Oppsett))]
297 X = [[] for _ in range(len(Oppsett))]
298 H = [[] for _ in range(len(Oppsett))]
299 V1 = [[] for _ in range(len(Oppsett))]
300 F_R0 = [[] for _ in range(len(Oppsett))]
301 F_L = [[] for _ in range(len(Oppsett))]
302 F_CL = [[] for _ in range(len(Oppsett))]
303 P_W = [[] for _ in range(len(Oppsett))]
304
305 for k in range(len(Oppsett)):
306     Gir = startgir
307     GIR[k].append(startgir)
308     t = 0
309     stegfaktor = 0.1/deltaT
310     v1 = v_start

```

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\Modell.py

```
311     stepcounter = 0
312     c = 0
313     u = 0
314     i = 0
315     p = 0
316     d = 0
317     x = x1
318     h = h1
319
320     while x < ll[len(ll) - 1]:
321         s = ss[c]
322         if x > ll[c]:
323             c = c + 1
324             s = ss[c]
325
326         stepcounter = stepcounter + 1
327
328         v1 = v1 / 3.6
329
330         if u == 10 * stegfaktor:
331             u = 0
332
333         if i == 100 * stegfaktor:
334             i = 0
335
336         if d == 100 * stegfaktor:
337             d = 0
338
339         if i != 0:
340             i = i + 1
341
342         if u != 0:
343             u = u + 1
344
345         if d != 0:
346             d = d + 1
347         else:
348             rpm = get_rpm_eff(Gir) * v1 * 3.6
349
350         # Gir = get_Gir_standard(Gir)
351
352         if k == 0:
353             Gir = get_Gir_Vuddu(Gir)
354         else:
355             Gir = get_Gir_standard(Gir)
356
357         rpm = get_rpm_eff(Gir) * v1 * 3.6
358         P_w = get_P_w(rpm)
359         GIR[k].append(Gir)
360
361         if Gir != GIR[k][stepcounter - 1]:
362             u = u + 1
363             i = i + 1
364             d = d + 1
365
366         # --- Girskift tidsbruk ---
367         if 0 < u <= 10 * stegfaktor:
368             P_w = P_w * 0
369         # --- Redusert effekt etter giring ---
370         if 10 * stegfaktor < i <= 100 * stegfaktor:
371             P_w = P_w * 1
372
```

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\Modell.py

```
373         # if d == 100 * stegfaktor and s < 10:
374         #     d = 0
375
376     # --- Varierende iri og mpd ---
377     # if IRI_MPD.P[p] < x < IRI_MPD.P[len(IRI_MPD.P)-1]:
378     #     p = p + 1
379     #
380     # if p > 71:
381     #     iri = 1.3
382     #     mpd = 1.22
383     #
384     # iri = IRI_MPD.IRI_list[p]
385     # mpd = IRI_MPD.MPD_list[p]
386
387     # --- Motstandskrefter ---
388     f_r = 0.00414 + 0.0000158 * iri * v1 + 0.00102 * mpd
389     # f_r = (k_r + k_s * v1)
390     # f_r = 0.015
391     F_ro = f_r * m * g * 1
392     F_l = 0.5 * rho * A * cw * (v1 + v0) ** 2
393     F_cl = m * g * 0.01 * s
394
395     F_R0[k].append(F_ro)
396     F_L[k].append(F_l)
397     F_CL[k].append(F_cl)
398
399     # if Gir == 10:
400     #     P_w = P_w*1
401
402     P_w = P_w*Oppsett[k]
403
404     P_w = P_w*1
405
406     # --- Konstant effekt ---
407     if k == 2:
408         P_w = eff_gammel * 1000 * 0.839
409     if k == 3:
410         P_w = eff_gammel * 1000 * 0.95
411     if k == 4:
412         P_w = eff_gammel * 1000 * 1
413     # -----
414
415     P_W[k].append(P_w)
416
417     # --- Sum av krefter ---
418     F_wn = F_ro + F_l + F_cl
419     F_wp = P_w / v1
420     F_ws = F_wp - F_wn
421
422     a = F_ws / m
423
424     t = t + deltaT
425     deltaX = v1 * deltaT + 0.5 * a * deltaT * deltaT
426
427     x = x + deltaX
428     deltaH = (deltaX * s) / 100
429
430     h = h + deltaH
431     v1 = v1 + a * deltaT
432
433     T[k].append(t)
434     X[k].append(x)
```

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\Modell.py

```
435         H[k].append(h)
436
437         RPM[k].append(rpm)
438
439         if v1 >= v_max/3.6:
440             v1 = v_max/3.6
441             print('Maks hastighet nådd')
442         else:
443             v1 = v1
444
445         v1 = v1 * 3.6
446         V1[k].append(v1)
447
448 # --- Printing -----
449 # print(stepcounter)
450 # print(iri)
451 # print(mpd)
452 # print(p)
453 # print('U =', u)
454 # print(i)
455 # print('Gir', Gir)
456 # print('kW', P_w / 1000)
457 # print('rpm', rpm)
458 # print('Luftmotstand', F_L)
459 # print('Stigningsmotstand', F_CL)
460 # print('Rullemotstand', F_RO)
461 # print('Negative krefter sum', F_WN)
462 # print('Positive krefter sum', F_WP)
463 # print('Sum krefter', F_WS)
464 # print('Tid totalt:', t)
465 # print('Akselerasjon:', a, 'm/s^2')
466 # print('Distanse:', x, 'm')
467 # print('Høyde økning:', deltaH, 'm')
468 # print('Høyde:', h, 'moh')
469 # print('Hastighet:', v1, 'km/t')
470
471 # --- Utregning av motstandsgraf -----
472
473 # Motstand_1 = F_CL
474 # Motstand_2 = [x + y for x, y in zip(F_CL, F_RO)]
475 # Motstand_3 = [x + y + z for x, y, z in zip(F_CL, F_RO, F_L)]
476 #
477 # Sum_motstand = [x + y + z for x, y, z in zip(F_CL, F_RO, F_L)]
478 # F_CL_DEL = [x/y for x, y in zip(F_CL, Sum_motstand)]
479 # F_RO_DEL = [x/y for x, y in zip(F_RO, Sum_motstand)]
480 # F_L_DEL = [x/y for x, y in zip(F_L, Sum_motstand)]
481 #
482 #
483 #
484 # Motstand_4 = [x * 100 for x in F_CL_DEL]
485 # Motstand_5 = [100 * (x + y) for x, y in zip(F_CL_DEL, F_RO_DEL)]
486 # Motstand_6 = [100 * (x + y + z) for x, y, z in zip(F_CL_DEL, F_RO_DEL, F_L_DEL)]
487
488 # --- Tilpasse steglengde VBOX- hastighet til beregnede x-koordinater -----
489
490 # mod_VBOX = len(speed_VBOX)/len(X[0])
491 # VS = []
492 # V1_eq = [[] for _ in range(len(Oppsett))]
493 #
494 # for hastighet in range(len(X[0])):
495 #     vs = speed_VBOX[math.floor(hastighet*mod_VBOX)]
496 #     VS.append(vs)
```

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\Modell.py

```
497 #
498 # for b in range(len(Oppsett)):
499 #     if len(V1[b]) < len(VS):
500 #         forhold = len(V1[b])/len(VS)
501 #         for r in range(len(VS)):
502 #             speed = V1[b][math.floor(r * forhold)]
503 #             V1_eq[b].append(speed)
504 #     else:
505 #         forhold = len(VS)/len(V1[b])
506 #         for r in range(len(VS)):
507 #             speed = V1[b][math.floor(r * forhold)]
508 #             V1_eq[b].append(speed)
509
510 Lengste = max(enumerate(V1), key = lambda tup: len(tup[1]))
511 Lengste_liste = Lengste[0]
512
513 mod_VBOX = len(speed_VBOX)/len(X[Lengste_liste])
514 VS = []
515 V1_eq = [[] for _ in range(len(Oppsett))]
516 X_eq = [[] for _ in range(len(Oppsett))]
517
518 for hastighet in range(len(X[Lengste_liste])):
519     vs = speed_VBOX[math.floor(hastighet * mod_VBOX)]
520     VS.append(vs)
521
522 for b in range(len(Oppsett)):
523     if len(V1[b]) < len(VS):
524         forhold = len(V1[b])/len(VS)
525         for r in range(len(VS)):
526             speed = V1[b][math.floor(r * forhold)]
527             V1_eq[b].append(speed)
528             distance = X[b][math.floor(r * forhold)]
529             X_eq[b].append(distance)
530     else:
531         forhold = len(VS)/len(V1[b])
532         for r in range(len(VS)):
533             speed = V1[b][math.floor(r * forhold)]
534             V1_eq[b].append(speed)
535             distance = X[b][math.floor(r * forhold)]
536             X_eq[b].append(distance)
537
538 # --- Plotting -----
539 # -----
540
541 #
542 # fig, ax1 = plt.subplots()
543 #
544 # color = 'tab:red'
545 # ax1.set_xlabel('Horizontal avstand [m]')
546 # ax1.set_ylabel('Hastighet modell [km/t]', color=color)
547 # ax1.plot(X, V1[0], color=color)
548 # ax1.tick_params(axis='y', labelcolor=color)
549 #
550 # ax2 = ax1.twinx() # instantiate a second axes that shares the same x-axis
551 # ax1.set_ylim(min(V1[0] + VS) - 1, max(V1[0] + VS) + 1)
552 # ax2.set_ylim(min(V1[0] + VS) - 1, max(V1[0] + VS) + 1)
553 # color = 'tab:blue'
554 # ax2.set_ylabel('Hastighet VBOX [km/t]', color=color) # we already handled the x-label with
    ax1
555 # ax2.plot(X, VS, color=color)
556 # ax2.tick_params(axis='y', labelcolor=color)
557 #
```


File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\Modell.py

```
558 # fig.tight_layout() # otherwise the right y-label is slightly clipped
559 #
560 # plt.grid()
561 # plt.show()
562
563 # fig, ax1 = plt.subplots()
564 #
565 # color = 'tab:red'
566 # ax1.set_xlabel('Horisontal avstand [m]')
567 # ax1.set_ylabel('Hastighet modell [km/t]', color=color)
568 # ax1.plot(X, V1, color=color)
569 # ax1.tick_params(axis='y', labelcolor=color)
570 #
571 # ax2 = ax1.twinx() # instantiate a second axes that shares the same x-axis
572 #
573 # color = 'tab:blue'
574 # ax2.set_ylabel('Hastighet VBOX', color=color) # we already handled the x-label with ax1
575 # ax2.plot(X, VS, color=color)
576 # ax2.tick_params(axis='y', labelcolor=color)
577 #
578 # fig.tight_layout() # otherwise the right y-label is slightly clipped
579 # plt.show()
580
581 # fig, ax1 = plt.subplots()
582 #
583 # color = 'tab:red'
584 # ax1.set_xlabel('Tid [s]')
585 # ax1.set_ylabel('Høyde [moh]', color=color)
586 # ax1.plot(T, H, color=color)
587 # ax1.tick_params(axis='y', labelcolor=color)
588 #
589 # ax2 = ax1.twinx() # instantiate a second axes that shares the same x-axis
590 #
591 # color = 'tab:blue'
592 # ax2.set_ylabel('Hastighet', color=color) # we already handled the x-label with ax1
593 # ax2.plot(T, V1[0], color=color)
594 # ax2.tick_params(axis='y', labelcolor=color)
595 #
596 # fig.tight_layout() # otherwise the right y-label is slightly clipped
597 # plt.grid()
598 # plt.show()
599
600 # fig, ax1 = plt.subplots(figsize=(10,4))
601 # alpha = 1
602 # linewidth = 2
603 # ax1.set_xlabel('Horisontal avstand [m]')
604 # ax1.set_ylabel('GIR')
605 # ax1.plot(X[0], GIR[0][:len(X[0])], alpha=alpha, linewidth=linewidth)
606 # ax1.plot(X[1], GIR[1][:len(X[1])], alpha=alpha, linewidth=linewidth)
607 # ax1.plot(X[2], GIR[2][:len(X[2])], alpha=alpha, linewidth=linewidth)
608 # ax1.plot(X[3], GIR[3][:len(X[3])], alpha=alpha, linewidth=linewidth)
609 # ax1.tick_params(axis='y')
610 #
611 # fig.tight_layout() # otherwise the right y-label is slightly clipped
612 # plt.show()
613 #
614 # fig, ax1 = plt.subplots(figsize=(10,4))
615 # alpha = 1
616 # linewidth = 2
617 # ax1.set_xlabel('Horisontal avstand [m]')
618 # ax1.set_ylabel('RPM')
619 # ax1.plot(X[0], RPM[0], alpha=alpha, linewidth=linewidth)
```

```

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\Modell.py
620 # ax1.plot(X[1], RPM[1], alpha=alpha, linewidth=linewidth)
621 # ax1.plot(X[2], RPM[2], alpha=alpha, linewidth=linewidth)
622 # ax1.plot(X[3], RPM[3], alpha=alpha, linewidth=linewidth)
623 # ax1.tick_params(axis='y')
624 #
625 # fig.tight_layout() # otherwise the right y-label is slightly clipped
626 # plt.show()
627
628 # --- Girsift og turtall -----
629
630 fig, ax1 = plt.subplots(figsize=(10,4))
631 alpha = 1
632 linewidth = 2
633 ax1.set_xlabel('Horisontal avstand [m]')
634 ax1.set_ylabel('GIR')
635 ax1.plot(X[0], GIR[0][:len(X[0])], alpha=alpha, linewidth=linewidth)
636 ax1.tick_params(axis='y')
637
638 fig.tight_layout() # otherwise the right y-label is slightly clipped
639 plt.show()
640
641 fig, ax1 = plt.subplots(figsize=(10,4))
642 alpha = 1
643 linewidth = 2
644 ax1.set_xlabel('Horisontal avstand [m]')
645 ax1.set_ylabel('RPM')
646 ax1.plot(X[0], RPM[0], alpha=alpha, linewidth=linewidth)
647 ax1.tick_params(axis='y')
648
649 fig.tight_layout() # otherwise the right y-label is slightly clipped
650 plt.show()
651
652 # --- Forskjellig motoreffekt -----
653
654 # fig, ax1 = plt.subplots()
655 #
656 # color = 'tab:pink'
657 # ax1.set_xlabel('Tid [s]')
658 # ax1.set_ylabel('Høyde [moh]', color=color)
659 # ax1.plot(T, H, color=color)
660 # ax1.tick_params(axis='y', labelcolor=color)
661 #
662 # ax2 = ax1.twinx() # instantiate a second axes that shares the same x-axis
663 #
664 # color = 'tab:blue'
665 # ax2.set_ylabel('RPM', color=color) # we already handled the x-label with ax1
666 # ax2.plot(T, RPM, color=color)
667 # ax2.tick_params(axis='y', labelcolor=color)
668 #
669 # fig.tight_layout() # otherwise the right y-label is slightly clipped
670 # plt.show()
671
672 # Hastighet, Stigning mot Avstand
673
674 # fig, ax1 = plt.subplots(figsize=(10,10))
675 # ax1.set_xlabel('Horisontal avstand [m]')
676 # ax1.set_ylabel('Hastighet [km/t]') # we already handled the x-label with ax1
677 # ax1.plot(X[0], V1[0], label='530')
678 # ax1.plot(X[1], V1[1], label='580')
679 # ax1.plot(X[2], V1[2], label='660')
680 # ax1.plot(X[3], V1[3], label='770')
681 # ax1.plot(X[4], V1[4], label='580 gammel')

```

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\Modell.py

```
682 # ax1.plot(X[4], VS, label='VB0X')
683 #
684 # ax1.set_ylim([27.5, 95])
685 # ax1.minorticks_on()
686 # ax1.tick_params(axis='y')
687 # plt.legend(loc=6, prop={'size': 10})
688 # # ax1.set_facecolor('#bfbfbf')
689 #
690 # color = 'blue'
691 # ax2 = ax1.twinx() # instantiate a second axes that shares the same x-axis
692 #
693 # ax2.set_ylabel('Høyde [m]', color=color)
694 # ax2.plot(X[0], H[0], color =color)
695 # ax2.tick_params(axis='y', colors=color)
696 # ax2.set_ylim([-5, 130])
697 # ax2.minorticks_on()
698 # ax2.grid(False)
699
700 # plt.savefig(save_results_to + 'Hastighet_ulik_effekt.pdf', bbox_inches='tight')
701 # fig.tight_layout() # otherwise the right y-label is slightly clipped
702 # plt.show()
703
704 # --- Sammenligning av ny og gammel graf -----
705
706 fig, ax1 = plt.subplots(figsize=(10,5))
707 ax1.set_xlabel('Horisontal avstand [m]')
708 ax1.set_ylabel('Hastighet [km/t]') # we already handled the x-label with ax1
709
710 # plt.axvline(220, color='grey')
711 # plt.axvline(680, color='grey')
712 # plt.axvline(1120, color='grey')
713 # plt.axvline(2130, color='grey')
714 #
715 # plt.axvline(390, color='c')
716 # plt.axvline(810, color='c')
717 # plt.axvline(1190, color='c')
718 # plt.axvline(1550, color='c')
719 # plt.axvline(2120, color='c')
720 #
721 # plt.text(220, 86, '12 > 11', bbox={'facecolor':'white','alpha':1,'edgecolor':'grey','pad':2
722 }, ha='left', va='bottom', rotation=45, size=9)
722 # plt.text(680, 86, '11 > 10', bbox={'facecolor':'white','alpha':1,'edgecolor':'grey','pad':2
723 }, ha='left', va='bottom', rotation=45, size=9)
723 # plt.text(1100, 86, '10 > 9', bbox={'facecolor':'white','alpha':1,'edgecolor':'grey','pad':2
724 }, ha='left', va='bottom', rotation=45, size=9)
724 # plt.text(2150, 86, '10 > 11', bbox={'facecolor':'white','alpha':1,'edgecolor':'grey','pad':
725 2}, ha='left', va='bottom', rotation=45, size=9)
725 #
726 # plt.text(390, 86, '12 > 11', bbox={'facecolor':'white','alpha':1,'edgecolor':'c','pad':2},
727 ha='left', va='bottom', rotation=45, size=9)
727 # plt.text(810, 86, '11 > 10', bbox={'facecolor':'white','alpha':1,'edgecolor':'c','pad':2},
728 ha='left', va='bottom', rotation=45, size=9)
728 # plt.text(1190, 86, '10 > 9', bbox={'facecolor':'white','alpha':1,'edgecolor':'c','pad':2},
729 ha='left', va='bottom', rotation=45, size=9)
729 # plt.text(1550, 86, '9 > 10', bbox={'facecolor':'white','alpha':1,'edgecolor':'c','pad':2},
730 ha='left', va='bottom', rotation=45, size=9)
730 # plt.text(2050, 86, '9 > 10', bbox={'facecolor':'white','alpha':1,'edgecolor':'c','pad':2},
731 ha='left', va='bottom', rotation=45, size=9)
731
732 # Forhold = len(V1[4])/len(V1[3])
733 # V3MOD = []
734 #
```

```

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\Modell.py
735 # for hastighet in range(len(V1[4])):
736 #     v3mod = V1[3][math.floor(hastighet*Forhold)]
737 #     V3MOD.append(v3mod)
738
739 ax1.plot(X_eq[0], VS, label='VB0X', color='g')
740 # ax1.plot(X_eq[1], V1_eq[1], label='R580 standard giring', color='r')
741 # ax1.plot(X_eq[0], V1_eq[0], label='R580 tilpasset giring', color='orange', linewidth=2)
742 ax1.plot(X[2], V1[2], label='R580 gammel modell justert', color='orange', linewidth=2)
743
744 ax1.plot(X_eq[3], V1_eq[3], '--', label='R580 gammel modell 95 %', color='c')
745 # ax1.plot(X[4], V1[4], '--', label='R580 gammel modell 85 %', color='orange')
746 # ax1.plot(X[4], V3MOD, '--', label='R580 gammel modell 80 %', color='orange')
747
748
749
750
751 plt.legend(loc=9, prop={'size': 10})
752
753 # ax1.fill_between(X[4], V3MOD, V1[4], facecolor='orange', alpha=0.15, interpolate=True)
754
755 ax1.set_ylim([27.5, 85])
756 ax1.tick_params(axis='y')
757 plt.xticks(np.arange(0, max(X[0]) + 100, 200))
758 plt.legend(loc=9, prop={'size': 10})
759
760 # plt.axvline(290, color='c')
761 #
762 # plt.text(290, 86, '11 > 9', bbox={'facecolor':'white', 'alpha':1, 'edgecolor':'c', 'pad':2},
763 #         ha='left', va='bottom', rotation=45, size=9)
764
765 # color = 'blue'
766
767 # ax2 = ax1.twinx() # instantiate a second axes that shares the same x-axis
768
769 # ax2.set_ylabel('Vertikal avstand [m]', color=color)
770 # ax2.plot(X[0], H[0], color =color)
771 # ax2.tick_params(axis='y', colors=color)
772 # ax2.set_ylim([-5, 130])
773 # ax2.grid(False)
774
775 # plt.savefig(save_results_to + 'Hastighet_Scania_0kstad_justert_utnyttelsesgrad.pdf',
776 #             bbox_inches='tight')
777
778 fig.tight_layout() # otherwise the right y-label is slightly clipped
779 plt.show()
780
781 # --- Forskjellig motoreffekt -----
782
783 # fig, ax1 = plt.subplots(figsize=(10,4))
784 # ax1.set_xlabel('Horisontal avstand [m]')
785 # ax1.set_ylabel('Hastighet [km/t]') # we already handled the x-label with ax1
786 # ax1.plot(X[0], V1[0], label='Volvo 540')
787 # ax1.plot(X[0], VS, label='VB0X')
788 #
789 # ax1.set_ylim([27.5, 95])
790 # ax1.minorticks_on()
791 # ax1.tick_params(axis='y')
792 # plt.legend(loc=6, prop={'size': 10})
793 # ax1.set_facecolor('#bfbfbf')
794 #
795 # color = 'blue'
796 # ax2 = ax1.twinx() # instantiate a second axes that shares the same x-axis
797 #
798 # ax2.set_ylabel('Høyde [m]', color=color)

```

```

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\Modell.py
795 # ax2.plot(X[0], H[0], color =color)
796 # ax2.tick_params(axis='y', colors=color)
797 # ax2.set_ylim([-5, 130])
798 # ax2.minorticks_on()
799 # ax2.grid(False)
800 #
801 # plt.savefig(save_results_to + 'Hastighet_ulik_effekt.pdf', bbox_inches='tight')
802 # fig.tight_layout() # otherwise the right y-label is slightly clipped
803 # plt.show()
804
805 # --- Absolutt motstandsgraf -----
806
807 # fig, ax = plt.subplots()
808 #
809 # ax.plot(X, Motstand_1, X, Motstand_2, X, Motstand_3, color='black', linewidth=0.5)
810 # ax.set_xlabel('Horisontal avstand [m]')
811 # ax.fill_between(X, 0, Motstand_1, facecolor='#24a65e', interpolate=True)
812 # ax.fill_between(X, Motstand_1, Motstand_2, facecolor='#246ba6', interpolate=True)
813 # ax.fill_between(X, Motstand_2, Motstand_3, facecolor='#24a6a0', interpolate=True)
814 # ax.set_title('Motstandskrefter')
815 #
816 # plt.savefig(save_results_to + 'Motstandskrefter.pdf', bbox_inches='tight')
817 # ax.grid(axis='y')
818 # plt.show()
819 #
820 # fig10, ax = plt.subplots()
821 #
822 # ax.plot(X, F_CL_DEL, X, F_R0_DEL, X, F_L_DEL, color='black')
823 # ax.fill_between(X, 0, F_CL_DEL, facecolor='#24a65e', interpolate=True)
824 # ax.fill_between(X, F_CL_DEL, F_R0_DEL, facecolor='#246ba6', interpolate=True)
825 # ax.fill_between(X, F_R0_DEL, F_L_DEL, facecolor='#24a6a0', interpolate=True)
826 # ax.set_title('fill between where')
827
828 # plt.show()
829
830 # --- Motstandsgraf -----
831 # --- Laget uten seaborn ----
832
833 # fig, ax = plt.subplots()
834 #
835 #
836 # twin1 = ax.twinx()
837 # twin2 = ax.twinx()
838 #
839 # fig.subplots_adjust(right=0.75)
840 #
841 # twin2.spines["right"].set_position(("axes", 1.2))
842 # # twin2.spines.right.set_position(("axes", 1.2))
843 #
844 # ax.set_xlim(0, X[len(X)-1])
845 # ax.set_ylim(0, 100)
846 # twin1.set_ylim(min(V1[0])-5, max(V1[0])+5)
847 # twin2.set_ylim(min(H)-5, max(H)+5)
848 #
849 # ax.plot(X, Motstand_4, color='black', linewidth=0.5)
850 # ax.plot(X, Motstand_5, color='black', linewidth=0.5)
851 # ax.plot(X, Motstand_6, color='black', linewidth=0.5)
852 # # ax.plot(X, Motstand_5, color='black', linewidth=0.5)
853 # # ax.plot(X, Motstand_6, color='black', linewidth=0.5)
854 # twin1.plot(X, V1[0], color="red")
855 # twin2.plot(X, H, color= "blue")
856 #

```

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\Modell.py

```
857 # ax.set_xlabel("Horisontal avstand [m]")
858 # ax.set_ylabel("Andel av total kjøremotstand [%]")
859 # twin1.set_ylabel("Hastighet [km/t]", color="red")
860 # twin2.set_ylabel("Høyde [m]", color="blue")
861 #
862 # ax.fill_between(X, 0, Motstand_4, facecolor='#24a65e', alpha=0.5, interpolate=True)
863 # ax.fill_between(X, Motstand_4, Motstand_5, facecolor='#246ba6', alpha=0.5, interpolate=True)
864 # ax.fill_between(X, Motstand_5, Motstand_6, facecolor='#24a6a0', alpha=0.5, interpolate=True)
865 # # ax.set_title('Motstandskrefter andeler')
866 #
867 # # Set axes colors
868 # ax.spines["left"].set_color("black")
869 # twin1.spines["right"].set_color("black")
870 # twin2.spines["right"].set_color("black")
871 #
872 # ax.minorticks_on()
873 # twin1.minorticks_on()
874 # twin2.minorticks_on()
875 # # y_ticks = np.arange(0, 1.0001, 0.1)
876 # # plt.yticks(y_ticks)
877 #
878 # ax.tick_params(direction="in", which="both", colors="black")
879 # twin1.tick_params(direction="in", which="both", colors="red")
880 # twin2.tick_params(direction="in", which="both", colors="blue")
881 #
882 # ax.set_title('Motstandskrefter, 111 kg/kW', fontsize = 12)
883 #
884 # # plt.text(1450, 34, '(a)')
885 # # plt.text(2250, 20, '(b)')
886 # # plt.text(2500, 62, '(c)')
887 # # plt.text(100, 1, '0,0%')
888 # # plt.text(445, 12, '3,5%')
889 # # plt.text(980, 48, '7,0%')
890 # # plt.text(1365, 65, '3,5%')
891 # # plt.text(2100, 71, '0,0%')
892 #
893 # plt.savefig(save_results_to + 'Motstandskrefter_andeler_2.pdf', bbox_inches='tight')
894 # # ax.grid(axis='y')
895 # plt.show()
896 # -----
897 # fig,ax1=plt.subplots(figsize=(10,5))
898 # ax1.set_xlabel('Horisontalavstand[m]')
899 # ax1.set_ylabel('Hastighet[km/t]')#wealreadyhandledthex-labelwithax1
900 #
901 # plt.axvline(220,color='grey')
902 # plt.axvline(680,color='grey')
903 # plt.axvline(1120,color='grey')
904 # plt.axvline(1740,color='grey')
905 # plt.axvline(2130,color='grey')
906 # plt.axvline(2300,color='grey')
907 #
908 # plt.axvline(390,color='c')
909 # plt.axvline(810,color='c')
910 # plt.axvline(1190,color='c')
911 # plt.axvline(1550,color='c')
912 # plt.axvline(2120,color='c')
913 #
914 # plt.text(220,86,'12>11',bbox={'facecolor':'white','alpha':1,'edgecolor':'grey','pad':2},ha
    = 'left',va='bottom',rotation=45,size=9)
915 # plt.text(680,86,'11>10',bbox={'facecolor':'white','alpha':1,'edgecolor':'grey','pad':2},ha
```

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\Modell.py

```
915 = 'left', va='bottom', rotation=45, size=9)
916 # plt.text(1100,86, '10>9', bbox={'facecolor':'white', 'alpha':1, 'edgecolor':'grey', 'pad':2}, ha
    = 'left', va='bottom', rotation=45, size=9)
917 # plt.text(1740,86, '9>10>9', bbox={'facecolor':'white', 'alpha':1, 'edgecolor':'grey', 'pad':2},
    ha='left', va='bottom', rotation=45, size=9)
918 # plt.text(2150,86, '9>10', bbox={'facecolor':'white', 'alpha':1, 'edgecolor':'grey', 'pad':2}, ha
    = 'left', va='bottom', rotation=45, size=9)
919 # plt.text(2300,86, '10>11', bbox={'facecolor':'white', 'alpha':1, 'edgecolor':'grey', 'pad':2}, ha
    = 'left', va='bottom', rotation=45, size=9)
920 #
921 #
922 # plt.text(390,86, '12>11', bbox={'facecolor':'white', 'alpha':1, 'edgecolor':'c', 'pad':2}, ha='
    left', va='bottom', rotation=45, size=9)
923 # plt.text(810,86, '11>10', bbox={'facecolor':'white', 'alpha':1, 'edgecolor':'c', 'pad':2}, ha='
    left', va='bottom', rotation=45, size=9)
924 # plt.text(1190,86, '10>9', bbox={'facecolor':'white', 'alpha':1, 'edgecolor':'c', 'pad':2}, ha='
    left', va='bottom', rotation=45, size=9)
925 # plt.text(1550,86, '9>10', bbox={'facecolor':'white', 'alpha':1, 'edgecolor':'c', 'pad':2}, ha='
    left', va='bottom', rotation=45, size=9)
926 # plt.text(2050,86, '10>11', bbox={'facecolor':'white', 'alpha':1, 'edgecolor':'c', 'pad':2}, ha='
    left', va='bottom', rotation=45, size=9)
927 #
928 #
929 # ax1.plot(X[0],V1[0],label='FH 540 ny modell',color='r')
930 # ax1.plot(X[1],V1[1], '--',label='FH 540 gammel modell',color='orange')
931 # ax1.plot(X[1],VS,label='VB0X',color='g')
932 # plt.legend(loc=9,prop={'size':10})
933 #
934 # ax1.set_ylim([27.5,85])
935 # ax1.tick_params(axis='y')
936 # plt.xticks(np.arange(0,max(X[0])+100,200))
937 # plt.legend(loc=9,prop={'size':10})
938 #
939 # color='blue'
940 # ax2=ax1.twinx()#instantiateasecondaxesthatsharethesamex-axis
941 #
942 # ax2.set_ylabel('Vertikalavstand[m]',color=color)
943 # ax2.plot(X[0],H[0],color=color)
944 # ax2.tick_params(axis='y',colors=color)
945 # ax2.set_ylim([-5,110])
946 # ax2.grid(False)
947 #
948 # plt.savefig(save_results_to+'Hastighet_ny_og_gammel_volvo_0kstadbakken.pdf',bbox_inches='
    tight')
949 # fig.tight_layout()#otherwisetherighty-labelisslightlyclipped
950 # plt.show()
951
952 # -----Diverse print konsoll-----
953
954 end = time.time()
955 print('Tid:', round(T[0][len(T[0])-1],2), 'v_slutt:', round(V1[0][len(V1[0])-1],1), 'v_max:'
    , round(max(V1[0]), 1), 'v_min:', round(min(V1[0]), 1))
956 print('Tid:', round(T[1][len(T[1])-1],2), 'v_slutt:', round(V1[1][len(V1[1])-1],1), 'v_max:'
    , round(max(V1[1]), 1), 'v_min:', round(min(V1[1]), 1))
957 # print('Tid:', round(T[2][len(T[2])-1],2), 'v_slutt:', round(V1[2][len(V1[2])-1],1), 'v_max
    :', round(max(V1[2]), 1), 'v_min:', round(min(V1[2]), 1))
958 # print('Tid:', round(T[3][len(T[3])-1],2), 'v_slutt:', round(V1[3][len(V1[3])-1],1), 'v_max
    :', round(max(V1[3]), 1), 'v_min:', round(min(V1[3]), 1))
959 # print('Tid:', round(T[4][len(T[4])-1],2), 'v_slutt:', round(V1[4][len(V1[4])-1],1), 'v_max
    :', round(max(V1[4]), 1), 'v_min:', round(min(V1[4]), 1))
960
961 average = mean(ss)
```

File - C:\Users\simen\PycharmProjects\pythonProject\Hastighetsmodell\Modell.py

```
962 average_pw1 = mean(P_W[1])
963 average_pw2 = mean(P_W[2])
964 print("Gjennomsnittlig stigning =", round(average, 2), "%")
965 print("Gjennomsnittlig effekt ny modell", round(average_pw1/1000,2))
966 print("Gjennomsnittlig effekt gammel modell", round(average_pw2/1000,2))
967 print("Ny modell prosent av gammel snitt", round(average_pw1/10/425,3), "%")
968
969 GEH_list = [[] for _ in range(len(Oppsett))]
970 GEH_snitt = []
971 for w in range(len(Oppsett)):
972     for q in range(len(V1_eq[w])):
973         GEH = (math.sqrt(2 * (V1_eq[w][q] - VS[q]) ** 2) / (V1_eq[w][q] + VS[q]))
974         GEH_list[w].append(GEH)
975     geh_snitt = sum(GEH_list[w])/len(GEH_list[w])
976     GEH_snitt.append(geh_snitt)
977     print('GEH_snitt', w, '=', GEH_snitt[w])
978
979 # --- Tid koden bruker på å kjøre -----
980 print(end-start)
981
```