

Deze master thesis is deel van een reeks werken rond de E-bike droom van Ir. Tomas Keppens. Na jarenlange samenwerking met KU Leuven wierp deze samenwerking met de Universiteit zijn vruchten af. In het schooljaar van 2016-2017 ontstond het eerste prototype van deze next-generation e-bike uit de masterproef van Jorrit Heidbuchel. Kort hierna richtte Tomas en Jorrit samen IntuEdrive op.

De dag van vandaag horen we niets anders dan problemen omtrent het file leed. Veel mensen nemen al snel de auto voor korte afstanden (<25 km). In deze auto's zit meestal enkel 1 persoon. Dit gepaard met de jaarlijkse groei van het Belgische wagenpark, zoals te zien in tabel ##, leidt tot steeds meer problemen. Hier zien we wel een trend ontstaan. Steeds minder mensen rijden met een diesel wagen, het aantal benzine wagens blijft lichtjes stijgen en er is een sterke groei in elektrische en hybride wagens, +41.1% en +39% respectievelijk. Hieruit kunnen we afleiden dat het file probleem er niet beter op wordt, wat ook te zien is in figuur ##, en dat de vraag naar milieu vriendelijk transport sterk aan het toenemen is.

Laat dit nu de problemen zijn die intuEdrive probeert op te lossen met hun snelle E-bike, een groen alternatief voor de auto. Hoe meer mensen een elektrische fiets gebruiken voor het woon-werkverkeer, hoe minder files er zullen zijn. Maar om meer mensen op de elektrische fiets te krijgen, moet deze aantrekkelijker worden. Zo zou een e-bike snel, veilig en intuïtief moeten zijn. Hier ontstaat echter wel een probleem. Snelheid en veiligheid spreken elkaar tegen. Hoe sneller de fiets, hoe minder reactietijd de fietser heeft om een ongeval te ontwijken. Hiervoor heeft IntuEdrive een oplossing, hun eigen innovatieve e-CVT systeem. Dit systeem heeft vele voordelen. Ten eerste verlaagt het systeem de remafstand met meer dan 60%. Dit gepaard met een stevig frame, kleinere wielen en dikkere banden zorgt ervoor dat deze fiets heel stabiel en veilig is. Hier bovenop heeft de fiets ook nog elektrische remmen en kan deze automatisch schakelen. Al deze eigenschappen zorgen ervoor dat de gebruiker meer aandacht kan besteden aan de weg in plaats van zich zorgen te maken over onbelangrijke zaken. Dit resulteert in twee keer meer ruimte om een ongeval te ontwijken en meer tractie in vergelijking met de traditionele elektrische fietsen. Een veilige en intuïtieve rijervaring maakt CoSaR: the Convenient, Safe and Reliable speed E-bike. Bovendien is het onderhoudsvrij en makkelijk te gebruiken. We kunnen dus zeggen dat CoSaR een goed alternatief levert voor de auto.

Op 1 augustus van het jaar	2014	2015	2016	2017	2018
Personenwagens	5.555.499	5.623.579	5.712.061	5.785.447	5.853.782
- rijdend op benzine	2.029.688	2.091.327	2.199.038	2.335.349	2.518.942
- rijdend op diesel	3.458.424	3.457.526	3.424.592	3.339.034	3.193.658
- rijdend op gas	22.051	18.967	17.238	15.965	15.500
- met elektrische motor	1.792	2.871	4.368	6.552	9.244
- hybride	23.444	32.151	44.364	63.740	87.012
- niet nader bepaald	20.100	20.737	22.461	24.807	29.426
Inwoners per personenauto op 1 augustus (d)	2,01	1,99	1,97	1,96	1,94

(tabel van het voertuigenpark in België 2014-2018)

<https://statbel.fgov.be/nl/themas/mobiliteit/verkeer/voertuigenpark#figures>



<https://www.statistiekvlaanderen.be/filezwaarte>

(evolutie van de file zwaarte in Vlaanderen 2012-2018. (Filezwaarte = file lengte \* duur))

## Doel van de thesis

Deze thesis werkt verder op het prototype geleverd door intuEdrive, met als doel het aantrekkelijker maken van hun e-bike. Dit door de cadans van hun e-bike in real time te voorspellen aan de hand van de toestand van de fiets. Alhoewel de fiets automatisch schakelt, kan de ingestelde cadans niet aangenaam zijn voor de fietser. Waardoor de gebruiker de concentratie op de weg kan verliezen wanneer hij zijn cadans gaat aanpassen. Dit kan leiden tot gevaarlijke situaties.

Het voornaamste probleem hier is dat de optimale cadans van een fietser varieert van persoon tot persoon. Sommige mensen trappen liever een hogere cadans, andere mensen trappen liever trager. Voor dit probleem zal een machine learning algoritme ontwikkeld worden dat de toestand die de fiets meet als input binnenkrijgt en hiermee een zo goed mogelijke voorspelling maakt. De voorspellingen moeten in real time op de fiets berekend worden. Het algoritme moet draaien op een Raspberry Pi, samen met het controleprogramma van de fiets. Hoe meer voorspellingen per seconde het systeem kan leveren, hoe beter. Het doel is om aan 10Hz de cadans aan te passen. Trager dan dit zou merkbaar kunnen zijn door de gebruiker, wat kan leiden tot een onaangename rijervaring. Er moet natuurlijk ook rekening gehouden worden met de veiligheid van de fietser. Zo mogen opeenvolgende voorspellingen niet veel van elkaar verschillen en moet er een limiet staan op het maximum aantal toeren dat voorspeld kan worden.

Naast het voorspellen van de cadans, moet de fietser ook kunnen aangeven dat de voorspelde cadans incorrect is. Zo zou de fietser op een knop moeten drukken om aan te geven dat het algoritme foute voorspellingen maakt. Er zullen twee knoppen beschikbaar zijn. 1 voor aan te geven dat de cadans te hoog is en 1 voor een te lage cadans. Bijvoorbeeld: de gebruiker fietst een berg op en trapt 70rpm. Dit vindt hij niet goed omdat hij liever sneller trapt, in plaats van veel kracht op de pedalen te zetten. Dus drukt hij op de knop om zijn cadans te verhogen. Achter de schermen leert het machine learning algoritme bij dat de fietser in de huidige situatie een hogere cadans wil. Even later zou het algoritme een hogere cadans moeten voorspellen. De voorspellingen in andere situaties zouden quasi hetzelfde moeten blijven.

Om deze doelstellingen te bereiken zal er eerst een simulatie opgezet worden die de fiets zo goed mogelijk benadert. De berekeningen hiervoor zijn geleverd door Jorrit en zullen gecodeerd worden in python. Vervolgens worden een aantal machine learning algoritmes getest om te zien of deze het probleem aankunnen. De algoritmes maken deel uit van 2 libraries: Keras en scikit learn.

Als performantie indicator wordt er de Mean Squared Error gebruikt. Om deze indicator te gebruiken moet er ook een notie zijn van een fietsersmodel. Een soort van actor die naar gelang van de gesimuleerde toestand van de fiets gaat beslissen wat de optimale cadans is. Simpel gezegd is het fietsersmodel een functie met als input de toestand van de fiets en output een "optimale cadans". Deze functie is speculatief en kan makkelijk vervangen worden. Ten tweede moet het algoritme kunnen bijleren met een kleine hoeveelheid data. Aanpassingen gemaakt door de fietser zijn van toepassing op de recente toestand van de fiets. Er kan dus maar een beperkte hoeveelheid aan data gebruikt worden van de laatste X aantal seconden. Ten slotte wordt er vereist dat het algoritme snel bijleert, zowel op vlak van performantie als het aantal learning cycles. Elke verandering van de gebruiker gebeurt liefst in real time. Hier zit echter wel wat speling op. En de veranderingen moeten een betekenisvolle impact hebben.

VUL HIER AAN (uittesten op echte data, op fiets zetten...)

## Huidige systeem

Het is als sinds 2011 de droom van Tomas om een perfecte E-bike te maken, wanneer Tomas nog werkte bij Toyota. Het idee hier was om een E-bike te bouwen gemengd met enkele karakteristieken van een auto. Zo gebruikt de e-bike een transmissie gelijkaardig aan die van de Toyota Prius hybrid. De implementatie gebruikt een Continue Variabele Transmissie, of kortweg CVT. Er zijn verschillende soorten CVT's: een mechanische, een hydraulische en een elektrische. De fiets gebruikt de elektrische variant hiervan. Deze gebruikt 2 motoren elk met hun eigen functie. De ene motor stuurt de trapcadans aan. De andere motor reguleert de hoeveelheid kracht dat op de pedalen gezet wordt. Samen zorgen deze motoren ervoor dat de fiets natuurlijk aanvoelt. De motoren worden aangestuurd door de controller. Deze is geprogrammeerd in C.

Het huidige prototype, is al enkele iteraties ouder dan het prototype van Jorrit zijn masterproef, maar in essentie is het nog steeds dezelfde fiets. Figuur ## toont een blokdiagram van het huidige systeem. De fietser zoekt op elk moment een bepaalde snelheid te halen, hier aangeduid met  $r$ . Deze kan variëren naar gelang de situatie waarin de fietser zich bevindt. Bijvoorbeeld wanneer de fietser bergop rijdt kan de gewilde snelheid trager zijn dan bergaf. De fietser oefent ook acties uit op de fiets. Deze acties, hier  $u$ , zijn: de hoeveelheid kracht uitgeoefend op de pedalen, of er al dan niet aangegeven wordt dat de voorspellingen aangepast moeten worden, en tot slotte het besturen van de fiets. Met het laatste zal geen rekening gehouden worden. De fiets heeft als output verschillende metingen. De voornaamste metingen zijn: de hoek van de trapas, de gemeten kracht op de pedalen, de snelheid van de fiets en de helling waarop de fiets zich bevindt. Deze output wordt gebruikt door de controller om de fiets verder aan te sturen. Deze controller draait op een Raspberry Pi.

Voorafgaand deze thesis zorgde de controller ervoor dat de cadans op een stabiel niveau blijft van 70 rpm. De controller doet dit door continu te motoren aan te spreken en indien nodig te schakelen. Als er geen versnellingen meer overblijven, wanneer de fietser bijvoorbeeld heel hard aan het trappen is, dan zal de cadans uiteindelijk boven de 70 rpm gaan. De cadans controller, wat in deze thesis uitgewerkt zal worden, moet deze instelling overpakken door de output van de fiets te analyseren.



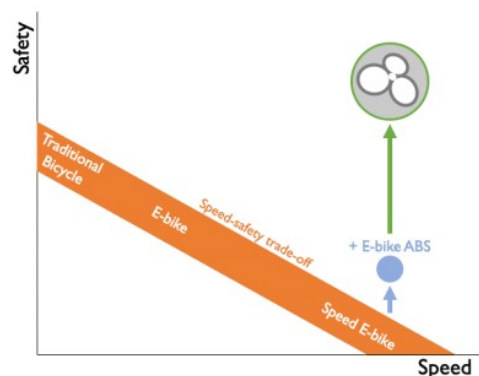
$R=[v\_ref]$   $u=[T_{cy}, u_c]$   $y=[\theta_{cr}, T_{cy,m}, v_{bike}, \alpha]$  (TODO: deftig afbeelden)

## IntuEdrive

Voor het ontstaan van intuEdrive werkte Tomas al meerdere keren samen met de Universiteit KU Leuven om zijn idee stap per stap uit te werken. Na jaren van samenwerking met de Universiteit, ontmoette Tomas Jorrit. Uit Jorrit zijn masterproef ontstond dan het eerste prototype, dat beter werkte dan verwacht. Vanaf dan besloten ze samen intuEdrive op te richten December 2017.



Het is misschien al duidelijk, maar intuEdrive maakt elektrische fietsen. Het is niet zomaar een standaard e-bike. De elektrische fiets geproduceerd door intuEdrive is een speed e-bike. Dit houdt in dat deze fiets een topsnelheid kan behalen van 45km/h, tegenover 25km/h van de normale pedelec. Maar met deze verhoging in snelheid komt natuurlijk een groter risico. Dankzij de implementatie van hun e-CVT systeem, gecombineerd met een stevig frame, kleinere wielen en dikkere banden maakt CoSaR: the Convenient, Safe and Reliable speed E-bike. Naast veiligheid, is CoSaR ook intuïtief te gebruiken. Met wat inspiratie van de one-pedal drive<sup>1</sup>, kan de fietser bijvoorbeeld remmen door de pedalen op een bepaalde positie stil te houden.



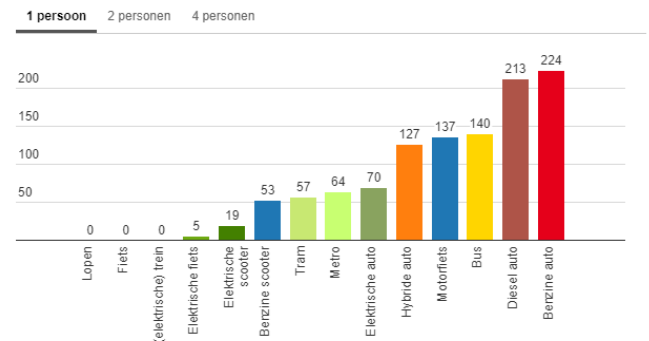
Waarom e-bikes? Ten eerste is er het file leed. Mensen pendelen vaak relatief korte afstanden (<25 km). En hiervoor nemen ze vaak de auto. Deze reis kan ook gemaakt worden met een e-bike. En soms zijn deze zelfs sneller met fiets dan met de auto. Voor de gebruikers die reeds een elektrische fiets gebruiken (<25km/h) zal de reistijd aanzienlijk verkorten. Het is ook wel handig dat deze speed e-bikes op het fietspad mogen rijden.

<sup>1</sup> one-pedal drive: <https://www.youtube.com/watch?v=577GaCWVpj0>

Ten tweede zijn e-bikes beter voor het milieu, ook al rij je met een elektrische auto. Fietsen wegen minder dan wagens en verstoken dus minder energie per kilometer.

Ten slotte is er natuurlijk nog het monetaire voordeel. E-bikes zijn een pak goedkoper dan de gemiddelde auto. De prijzen variëren van 1500 euro voor normale e-bikes (25 km/h), tot 5500-10000 euro voor speed e-bikes. Waarbij CoSaR zal geprijsd worden aan 6000 euro. Bovenop de voordelige prijs, is er ook nog het belastingvoordeel voor bedrijven en de fietsvergoeding voor woon-werkverkeer.

(<https://www.milieucentraal.nl/duurzaam-vervoer/fiets-ov-of-auto/>)



## Hoofdstuk 1: Simulatie

### Waarom een simulatie voor en nadelen

Zoals reeds aangehaald wordt er een simulatie gemaakt die de toestand van de fiets zo goed mogelijk probeert te benaderen. Om de werkdruk reëel te houden is de simulatie beperkt tot hetgeen wat nodig is. Zo zal er geen rekening gehouden worden met het manoeuvreren van de fiets, wind tegen of mee. Er zal ook enkel de toestanden berekend worden die gemeten worden door de fiets. Het is immer onnodig om data te verzamelen waar de fiets geen weet van heeft.

Het voordeel van de fietssimulatie is de enorme flexibiliteit. Uren aan data kan in een moment tijd gegenereerd worden, waardoor het makkelijk is om verschillende tests uit te voeren. Er is wel een hoop gelogde data beschikbaar. Maar deze heeft veel nul waarden, aangezien er niet constant door gefietst kan worden. Hierdoor is het moeilijk om een lange reeks van opeenvolgende data punten te verkrijgen. Bovendien kan deze data een bias hebben voor een bepaalde route. Waardoor we niet genoeg data hebben in verschillende omstandigheden. Met de fietssimulatie kunnen makkelijk verschillende situaties gegenereerd worden, waardoor we over enorm gevarieerde data kunnen beschikken.

Het voornaamste nadeel is dan toch wel de onnatuurlijke consistentie van de fietssimulatie. De geprogrammeerde actor is zeer consistent met de hoeveelheid vermogen dat hij levert. Er kan wel wat ruis toegevoegd worden, of zelfs een dominant been. Maar er is toch nog wel een verschil met een echte fietser. Zo zullen opeenvolgende cycli van de simulatie niet ver uit elkaar liggen en een trend vormen. Deze trend is bijvoorbeeld het versnellen van de cadans. Een menselijke fietser daarentegen trapt inconsistent. Er zal bijvoorbeeld wel een gelijkaardige trend ontstaan, maar opeenvolgende cycli tonen een eerder inconsistente versnelling of vertraging.

### blokdiagram van formules, cadans in functie van snelheid

#### modelleren van trapkracht = sinus (foto sinus tov echte cadans), reëel?

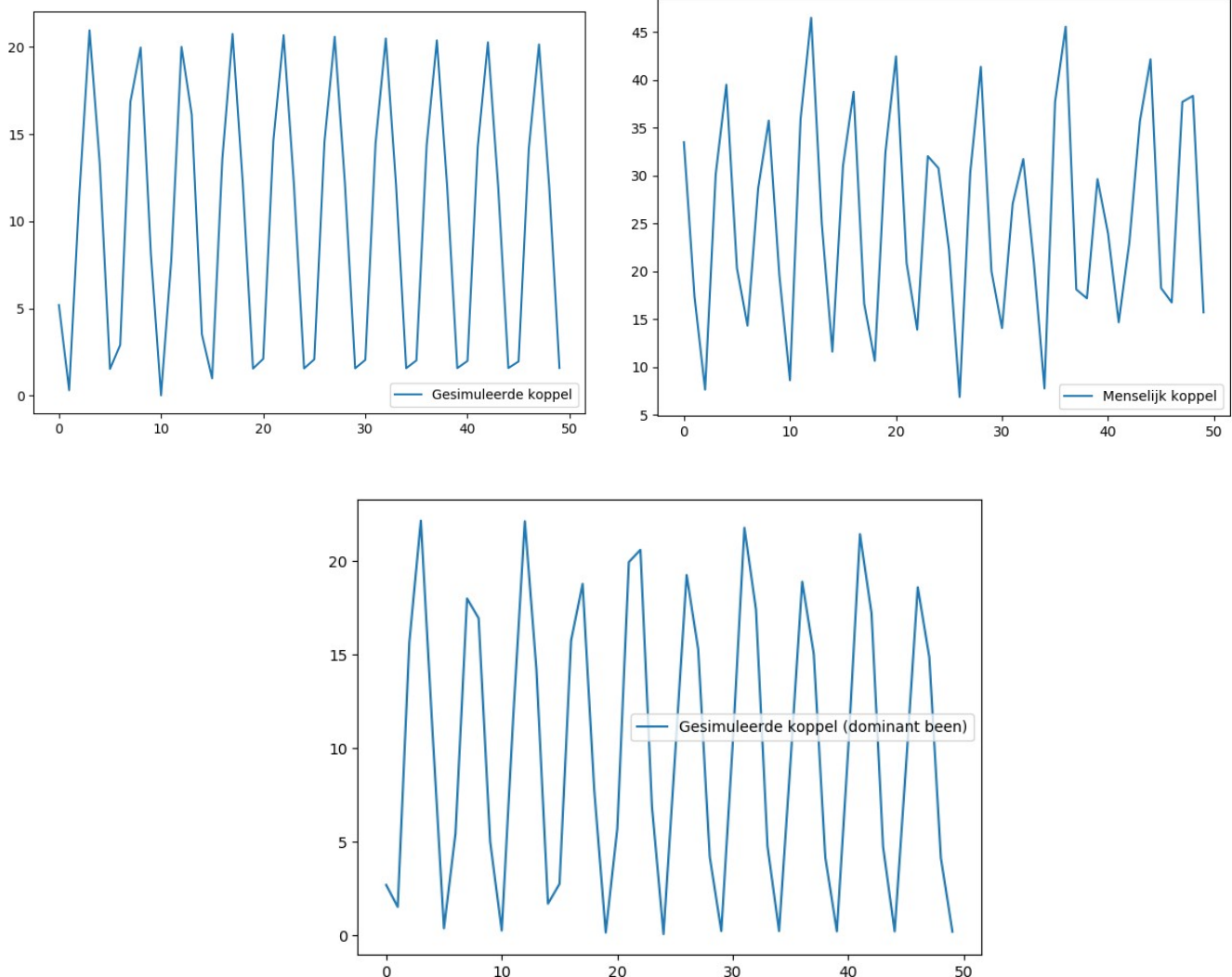
De modellering van het koppel is in functie van het gemiddelde koppel en een sinus functie. We gaan ervan uit dat 1 cyclus een volledige omwenteling is van de trapas. Het gegenereerde koppel vormt 2 pieken, 1 piek voor elk been. De functie ziet er als volgt uit:

$$t_{cy} = t_{dc} * (1 + \sin(2\theta - \pi/6))$$

Het gemiddelde koppel geleverd door de fietser wordt gemodelleerd als een proportionele regelaar. Het doel is om een bepaalde snelheid,  $v_{ref}$ , te behalen. Hoe verder de simulatie af is van deze doelstelling, hoe meer kracht er zal geleverd worden. Als deze snelheid overschreden is, dan wordt er gefreewheeled. Er wordt dus geen koppel meer geleverd. Om de kracht van de actor te limiteren, wordt er ook een maximum koppel ingesteld naar gelang de huidige cadans. De agressiviteit dat de actor doorvoert om de snelheid te halen wordt hier bepaald door  $K$ . De formules zien er als volgt uit:

$$t_{dc\_max} = (-\omega_{crank}[h - 1]) / 2 + 60$$
$$t_{dc} = \min(t_{dc\_max}, \max(0, -K * (v_{fiets}[h - 1] - v_{fiets\_ref})))$$

Hoe realistisch is deze modellering juist? Figuren ## en ## tonen het gesimuleerde en menselijk koppel respectievelijk beide aan 10Hz. Zoals te zien valt is het gesimuleerde koppel enorm consistent. Het menselijk koppel volgt duidelijk een cyclische functie, maar toont vormen van inconsistentie. Merk wel op dat er telkens een afwisseling is van een hoge en een lage piek. Dit wijst op een dominant been. Figuur ## toont een gesimuleerde dominant been. We stellen hier vast dat de gekozen modellering van het koppel realistisch genoeg is.



**waarom een goed lastmodel nodig? Geen wind tegen? Hypothese: cadans in functie van last**

**Terrein generatie**

**fietsmodel,fcc**

**verschillende mogelijkheden, plug and play**