

## Verkeerssimulatie



Namen: Storm Joannes, Charlie Choffat, Brandon Betz

Klas: V2B

Einddatum: 17 dec 2020

## Inhoudsopgave:

<b>Abstract</b>	<b>3</b>
<b>Onderwerp</b>	<b>3</b>
<b>Onderzoeksvragen</b>	<b>4</b>
<b>Vooronderzoek</b>	<b>4</b>
<b>Plan van aanpak</b>	<b>5</b>
<b>Experiment design</b>	<b>8</b>
<b>Resultaten experiment</b>	<b>8</b>
<b>Conclusie</b>	<b>8</b>
<b>Discussie</b>	<b>8</b>
<b>Bronnen</b>	<b>8</b>

# Abstract

## Onderwerp

Het onderwerp van dit verslag is het simuleren van een autoweg. Deze simulatie moet een ABMS zijn. Dat betekent agent-based modelling and simulation. Deze simulatie moet zo gemaakt worden dat we met behulp van de simulatie beide onderzoeksvragen kunnen beantwoorden. We waren begonnen met het maken van één rijstrook waar de auto's van rechts naar links bewegen en stopte wanneer het in de buurt kwam van een andere auto. Deze code hebben we met netlogo geschreven. Netlogo is een tool waarmee je makkelijk dingen kan visualiseren. We moeten ervoor zorgen dat auto's bepaalde acties kunnen uitvoeren, zoals versnellen en vertragen. Als basiscode hebben we de Nagel-Schreckenberg model gebruikt. Deze code moet natuurlijk worden uitgebreid zodat we nog bepaalde omgevingsveranderingen kunnen maken om het te laten werken met de onderzoeksvragen.

# Onderzoeksvragen

## **Onderzoeksvraag 1:**

Wat is het effect van de maximumsnelheid van de weg op de doorstroom van de weg?

## **Hypothese 1:**

Een hogere maximumsnelheid van de weg zal leiden tot een betere doorstroming over de weg omdat auto's sneller de weg op komen en de weg af gaan.

## **Onderzoeksvraag 2:**

Welke invloed heeft een kruispunt met stoplichten en een kruispunt zonder stoplichten (gelijkwaardig kruispunt) op de doorstroming, en hangt dit ook af van de verkeersdichtheid?

## **Hypothese 2:**

Een stoplicht zal succesvoller zijn voor een goede doorstroom bij een hoge verkeersdruk. Een gelijkwaardig kruispunt zal succesvoller zijn bij een lage verkeersdruk.

# Vooronderzoek

Om onze simulatie zo goed mogelijk te maken gaan we eerst kijken naar andere simulaties. Hiermee kunnen we zien hoe zij bepaalde gebeurtenissen simuleren en waarom ze het op die specifieke manier doen.

## **Traffic grid (netlogo):**

Netlogo heeft zelf meerdere voorbeeld simulaties waar gebruikers naar kunnen kijken voor inspiratie of waar ze zelf aanpassingen op kunnen maken. De traffic grid is zo een simulatie. Deze simulatie heeft meerdere kruispunten waar constant stoplichten wisselen van kleur. Vervolgens tonen ze de gemiddelde wachttijd, snelheid en de hoeveelheid auto's die stoppen. Wat bij deze simulatie interessant is is hoe ze werken met de stoplichten. Er wordt door de agents (de auto's) gekeken naar of het vlak voor hun rood is en als dat zo is stoppen ze. Als het vak vervolgens groen is mogen ze door. Het is heel simpel, maar het constant switchen van stoplichten en de auto's hier constant naar laten kijken zodat ze niet per ongeluk door rood rijden zou als je het niet goed aanpakt problemen opleveren.

Deze simulatie geeft een al een goed idee van hoe we aan de slag gaan. We maken een simpele kruispunt waar 4 stoplichten op komen waar auto's op moeten letten om te kunnen rijden. We kunnen vervolgens met hetzelfde kruispunt een gelijkwaardig kruispunt maken waar auto's dus niet op de stoplichten letten. We konden alleen

helaas geen project/simulatie hierover vinden dus de verkeersregels hierbij toepassen wordt nog interessant.

## Plan van aanpak

### Plan van aanpak:

- **GUI:**  
We zullen meerdere variabelen aanpasbaar maken door middel van bijvoorbeeld sliders (zoals de acceleratie, deceleratie, aantal auto's en bepaalde specs van de agents). Dit maakt het makkelijk om verschillende simulatie te runnen. Uiteraard zal de GUI buttons bevatten om de simulatie te runnen en ongeveer de helft van het beeld zal de visualisatie van de simulatie zijn.
- **Tijd:**  
De tijd kunnen we updaten in stappen of ticks en dan ervoor zorgen dat bijvoorbeeld bij elke tick bepaalde dingen gecontroleerd worden (zoals snelheid en afstand met andere agents) en op basis daarvan de actie bepalen. Zo blijft de simulatie constant geupdate. We hopen de tijd symmetrisch te maken, maar dat is lastig met netlogo. Netlogo zegt zelf dat het niet mogelijk is, maar er zijn voorbeeldcodes waar dit wel gebeurt. De berekeningen zijn asymmetrisch maar de visualisatie is symmetrisch.
- **Staat:**  
We kunnen kijken wat de staat van bepaalde waardes zijn, zoals de verkeersdichtheid, gemiddelde snelheid op de weg, en hoeveel auto's bij een bepaald punt zijn gekomen. Dit kan bij bepaalde tools makkelijk weergegeven worden met bijvoorbeeld monitors. Bij sommige tools is het moeilijk om met behulp van een kliksysteem de huidige staat van een agent te tonen dus of we dat gaan doen is nog niet zeker.
- **Resultaat:**  
We kunnen de resultaten van de simulaties overzetten in apart bestand. In dit bestand komen meerdere waardes (bijvoorbeeld de gemiddelde snelheid en aantal auto's) en met behulp van dit kunnen we de resultaten over meerdere simulaties analyseren om onze onderzoeksvragen te beantwoorden.
- **Modules:**  
Een snelweg hebben we nodig om te testen wat de doorstroming doet op een weg waar je 100 km/h mag in vergelijking met een normale weg waar je 50 km/h mag.

De gewone weg hebben we ook nodig om te meten wat stoplichten en kruispunten doen bij verschillende verkeers dichtheden, en welke de voorkeur zou hebben wanneer.

## **Welke tool gaan we gebruiken?**

Voor en nadelen elke tool:

### **Netlogo:**

- Voordelen:
  - Agents zijn makkelijk te maken
  - Syntax is makkelijk
  - Omgeving kan gemaakt worden met patches
  - Tijd kan goed bijgehouden worden
  - Snel
- Nadelen:
  - Moeilijk om de speciale situaties (rotondes, verkeersregels, in en uitvoegstrook) in te simuleren
  - Weinig ervaring met externe data
  - Omgeving moet niet te ingewikkeld zijn

### **Mesa:**

- Voordelen:
  - Agents zijn makkelijk te maken
  - Omgeving is makkelijk te maken
  - Goed om data te collecten
  - Maakt gebruik van Python
- Nadelen:
  - Lastig om een weg te visualiseren

### **Unity:**

- Voordelen:
  - duidelijke omgeving te visualiseren.
  - makkelijk om te gaan met speciale situaties, bijvoorbeeld een rotonde visualiseren.
- Nadelen:
  - lastig in gebruik.
  - wil je code werkende krijgen en testen moet je ook de hele visualisatie af hebben.
  - Meeste hebben nog niet met C# en unity gewerkt.

**SF(A):**

De ratings zijn 1 t/m 10 (1 laagste , 10 hoogste)

Suitability, feasibility en Modules zijn de totale scores van de onderwerpen eronder.

	Netlogo	Mesa	Unity
<b>Suitability</b>	<b>32</b>	<b>32</b>	<b>33</b>
Modules	20	18	21
- Agents	8	7	6
- Omgeving	7	7	8
- Speciale situatie	5	5	7
Efficiëntie	7	6	5
Compatibiliteit	5	9	7
<b>Feasibility</b>	<b>17</b>	<b>15</b>	<b>10</b>
Omgaan met tool?	9	7	5
Technisch haalbaar	8	8	5
<b>Totaal</b>	<b>49</b>	<b>47</b>	<b>43</b>

Kijken naar de voor en nadelen en naar het SFA zijn we op de conclusie gekomen om netlogo te gebruiken. De syntax is makkelijk te leren, snel en niet al te ingewikkeld. Het enige grote nadeel is dat het simuleren van speciale situaties een probleem kan zijn.

## Experiment design

Ons design zal voor onderzoeksvraag één anders zijn dan bij de tweede. Bij onderzoeksvraag 1 is het de bedoeling dat we 2 wegen boven elkaar hebben. Dit doen we zodat je direct de verschillen kan zien in plaats van de waardes opslaan en dan vervolgens vergelijken. Bij zo een simpele vraag als dit leek dat ons onnodig. Bij weg 1 zullen we een maximum snelheid hebben van 0.5, en bij weg 2 een maximumsnelheid van 1. De reden dat we deze waardes kiezen is omdat netlogo in patches werkt en de snelheid waarmee we dus bewegen is 1 (of 0.5) patch per tick. Zo gaat de auto bij elke tick een patch naar voren. Hier kunnen wij vervolgens een grafiek op toepassen en zien welke weg de beste doorstroom heeft.

Voor onderzoeksvraag 2 maken wij een kruispunt van 4 wegen met stoplichten en een zonder stoplichten. Deze twee zetten wij naast elkaar en testen we met



verschillende verkeers dichtheden. Hier kunnen wij ook weer een grafiek op zetten zodat je kan zien bij welke verkeersdichtheid welke beter is. Er zal een switch zijn waarmee je de werking van de stoplichten aan of uit kan zetten.

De stoplichten gaan werken met een timer (net als in het voorbeeld van het vooronderzoek). Deze zal 200 ticks duren zodat er een redelijke grote groep auto's tegelijk kan rijden. Als je dit veel hoger zou maken zou het te lang duren en te laag maakt het weer te kort.

Voor beiden onderzoeksvragen geldt, je laat beide scenario's met dezelfde verkeersdichtheid starten. De simulatie zal oneindig doorgaan totdat de gebruiker het zelf stopt. De auto's kijken een paar patches voor hun en als er niks in hun kijk range is versnellen ze op basis van een waarde die wij geven (en die je kan aanpassen). Als er een auto voor zich rijdt die slomer is verslomen ze zich op basis van de auto voor zich zodat er geen botsing ontstaat.

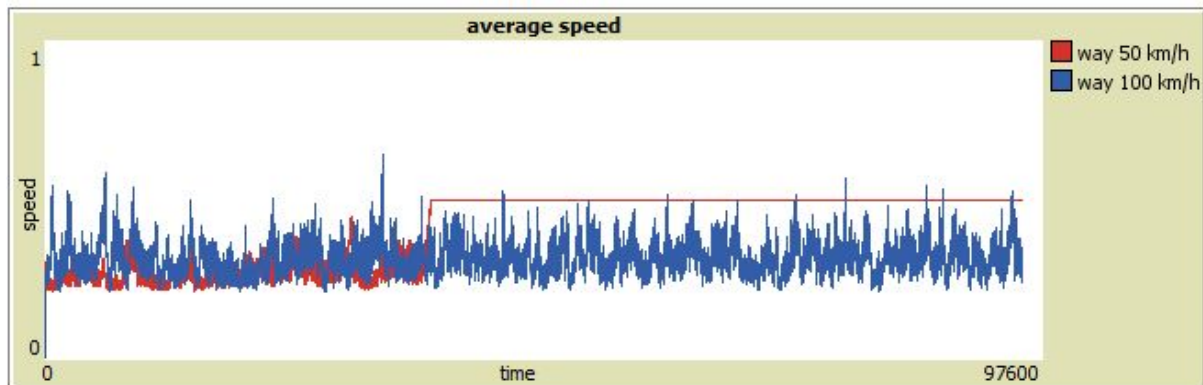
De doorstroming wordt gezien als hoe soepel het verkeer verloopt en wordt beoordeeld met 2 waardes: een gemiddelde snelheid per x aantal ticks en een line count. Als de gemiddelde snelheid of verhoogt of gelijk blijft is dat positief voor de doorstroming, want deze is dan of gelijk of wordt beter. De line count geeft aan hoe vaak auto's over een bepaald punt zijn gereden. Dit punt zal de locatie zijn van de stoplichten. Om de zoveel ticks wordt er gekeken naar het aantal en hiermee wordt er gekeken of er een verandering is. Net als bij de gemiddelde snelheid is het positief als de line count een stijging heeft of gelijk blijft.

Om verschillende resultaten te krijgen zullen we voor bepaalde waardes sliders maken zodat deze makkelijk aangepast kunnen worden. De code zal zo reproduceerbaar mogelijk gemaakt worden door middel van functies en comments zodat andere mensen er ook gewoon mee zouden kunnen werken zonder 100% van alle code te snappen.

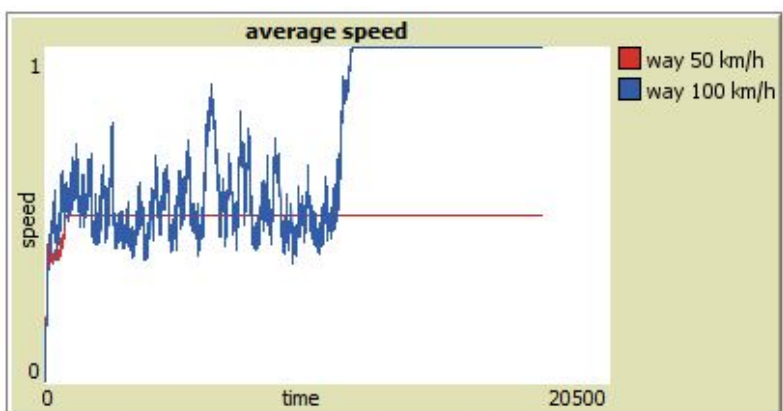
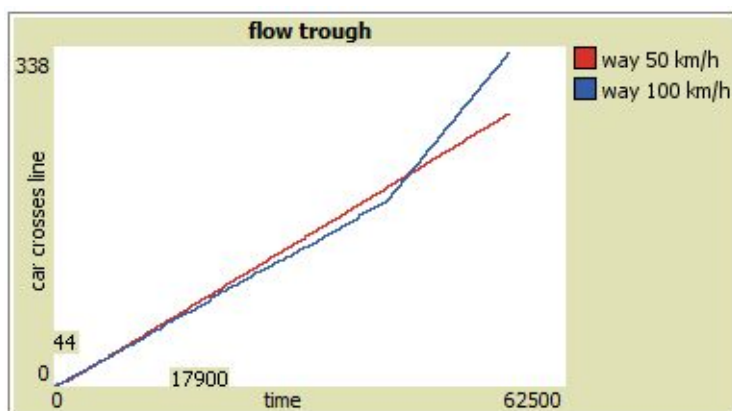
Wat het testen van onderzoeksvraag 2 betreft, gebruiken wij 3 verschillende sets aan waardes. De eerste set bevat een aantal auto's van 12 op de 4 lanes (wat een lage verkeersdichtheid moet vertegenwoordigen). De tweede set bevat een aantal auto's van 24 op de 4 lanes (wat wij beschouwen als hoge verkeersdruk). De derde set die wij gebruiken bevat een aantal auto's van 18 op de 4 lanes (dit zien wij als middelmatige verkeersdruk). Bij alle sets gebruiken wij een acceleratie van speed 0,0056 en een declaration van 0,055. Deze waardes hebben wij besloten door realistisch te kijken naar onze simulatie en dan te bepalen wat de beste en meest realistische resultaten gaf.

# Resultaten experiment

## Onderzoeksvraag 1:



Je kunt zien dat de snelheid op de weg met hogere maximum snelheid instabieler is. De beide wegen gebruiken dezelfde acceleratie en deceleratie grootte. Je zult zien dat als er een probleem in de doorstroming ontstaat de weg met een lagere maximumsnelheid dus eerder weer op een stabiele maximumsnelheid van 50 km/h zit. Je kan ook zien dat de weg met een dubbele maximumsnelheid vaak onder de 50 km/h komt. De doorstroming hier is dus bij de weg met lagere maximumsnelheid beter zodra er een "file" op beide wegen ontstaat. Als iedereen normaal kan doorrijden zul je wel zien dat je bij 100 km/h veel meer auto's sneller van a naar b kunt krijgen omdat het een hogere doorstroming heeft. Wel zie je ondanks dat dat de weg met een hoge snelheid vele malen langer erover doet om het fileprobleem op te lossen, maar zodra het probleem is opgelost de doorstroming wel weer veel hoger is dan die van de weg met lage maximum snelheid.



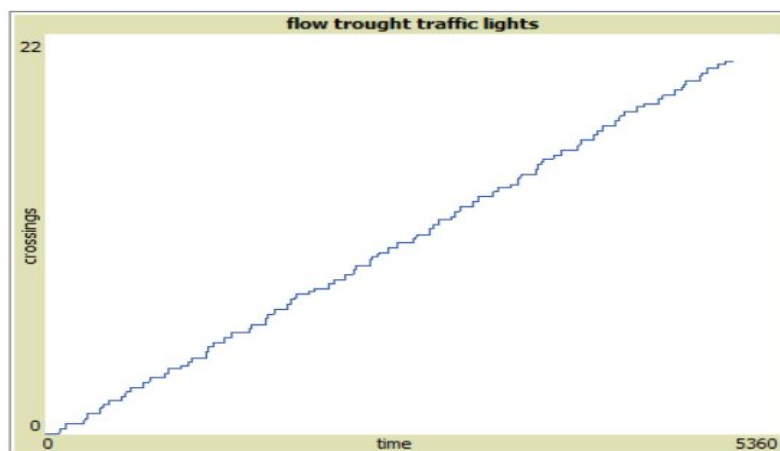
## Onderzoeksvraag 2:

Voor deze vraag hebben we 3 tests gerund. Eentje met 6 auto's op elke baan (24 totaal), een met 4 op links en beneden en 5 op rechts en boven (18 totaal) en een met 3 auto's op elke baan. Alle tests worden gestopt bij de 5000 ticks.

## 6 auto's:

- Stoplichten:

De stoplichten doen het zoals verwacht er goed bij een drukke baan. Alle auto's krijgen hun beurt om door te rijden wat zorgt voor een goede doorstroming. Als je ook kijkt naar de grafiek zie je dat de doorstroming van het verkeer ook een gelijke tempo heeft.

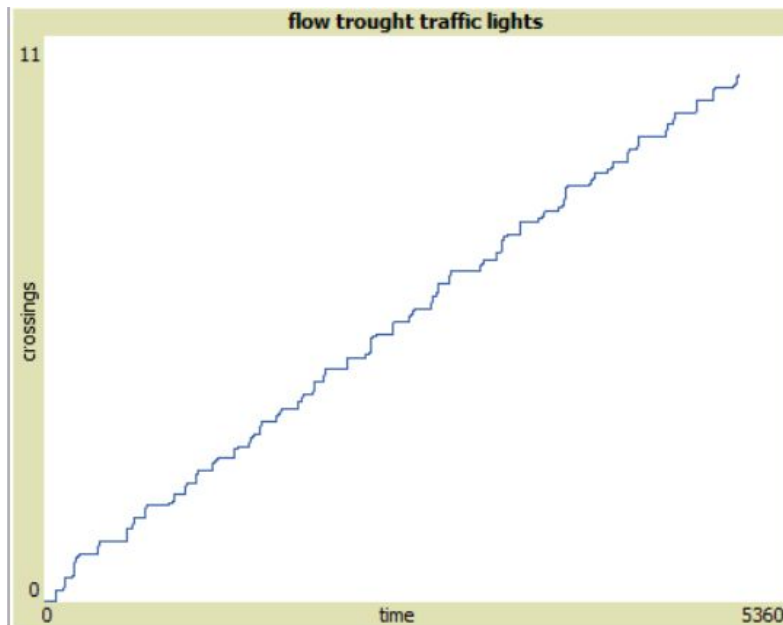


Als we kijken naar de verschillende staten die onze simulatie heeft opgeslagen zie je ook dat na de eerste staat (het opstarten) de gemiddelde snelheid constant blijft. Ook zie je dat de line count (wat aangeeft hoe vaak een auto over een bepaald punt is gegaan) een ongeveer gelijke stijging heeft van 5.

average speed,	line count,	total cars,	speed limit,	accelaration,	descelaration
0.00513333333333331,	0,	24,	0.6,	0.0056,	0.055
0.30046666666666666,	3.3829999999999982,	24,	0.6,	0.0056,	0.055
0.30046666666666666,	8.242999999999995,	24,	0.6,	0.0056,	0.055
0.30046666666666666,	12.556399999999991,	24,	0.6,	0.0056,	0.055
0.30046666666666666,	18.298999999999992,	24,	0.6,	0.0056,	0.055

- Gelijkwaardig kruispunt:

Het kruispunt doet het zoals ook al verwacht minder goed dan de stoplichten bij een drukke weg. Ondanks dat de doorstroming redelijk gelijk loopt (als je kijkt naar de line count), zie je dat er een momenten van opstoppingen zijn en dat de gemiddelde snelheid ook lager is.



De doorstroming heeft in vergelijking met de stoplichten vaker een file wat zorgt voor de langere rechte lijnen in de grafiek.

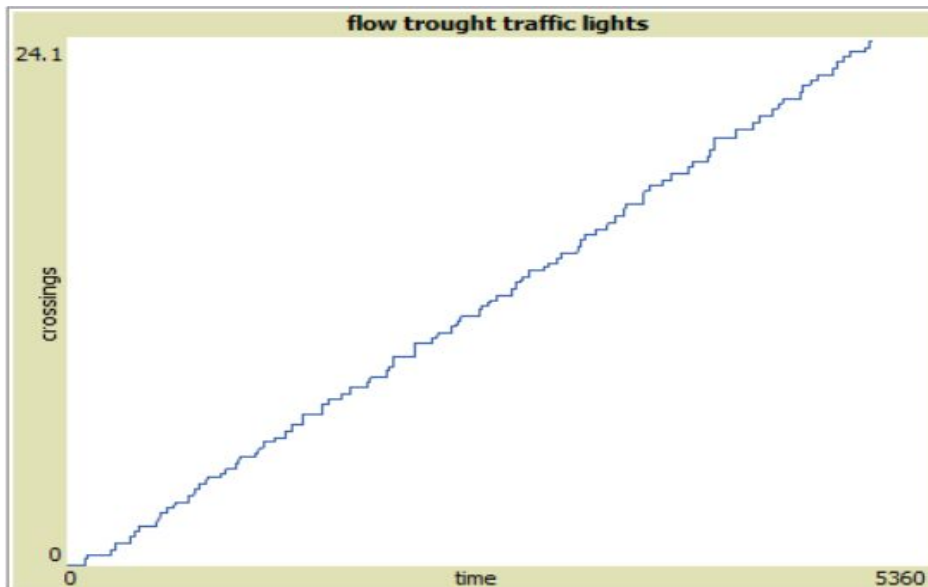
Kijkend naar de staten zie je dat de line count net als de stoplichten een ongeveer gelijke stijging heeft. Alleen wat je dan ziet is dat het een stijging is van ongeveer 2-3 wat dus staat voor een slechtere doorstroming, ook zie je dat de average speed een stuk lager is dan bij de stoplichten.

average speed,	line count,	total cars,	speed limit,	accelaration,	descelaration
0.00489999999999997,	0,	24,	0.6,	0.0056,	0.055
0.12117499999999994,	1.9207999999999996,	24,	0.6,	0.0056,	0.055
0.11784999999999995,	3.9591999999999999,	24,	0.6,	0.0056,	0.055
0.15174999999999997,	6.8991999999999998,	24,	0.6,	0.0056,	0.055
0.11344999999999993,	8.853599999999995,	24,	0.6,	0.0056,	0.055

#### 4-5 auto's:

##### - Stoplichten

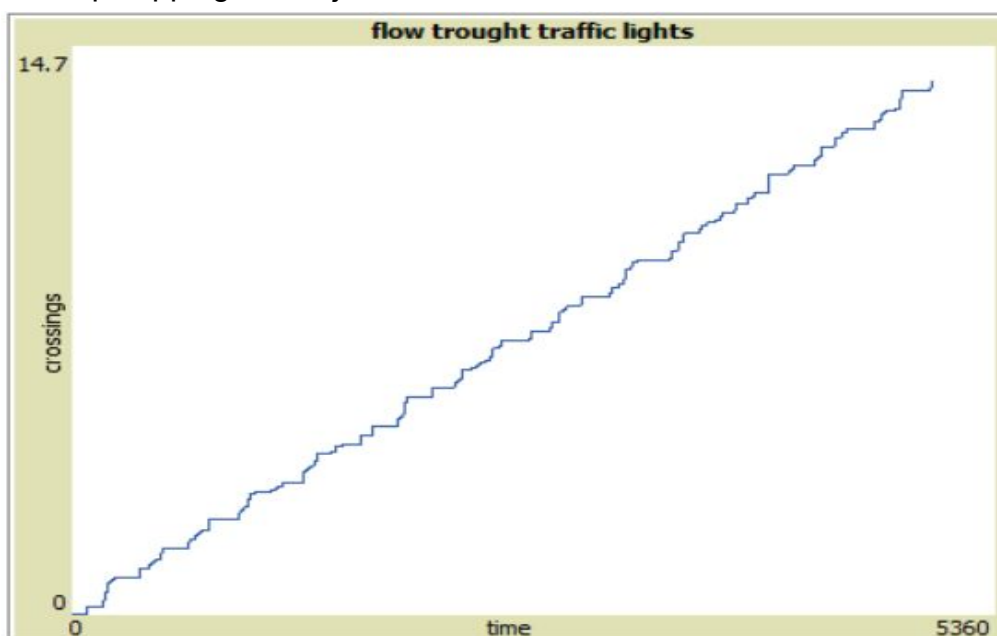
Net als bij de 6 auto's loopt de simulatie gewoon soepel. Auto's hebben genoeg tijd om over te steken en de andere auto's hoeven niet al te lang te wachten. De grafiek van de doorstroming leek dus ook ongeveer gelijk met hier en daar een paar kleine uitschieters die waarschijnlijk ontstaan waren door een paar auto's die te hard reden en dus ook hard moesten remmen.



Zoals je ziet is de average speed ook bijna gelijk en de line count heeft een stijging die iets lager is (ongeveer 4 in plaats van 5).

average speed,	line count,	total cars,	speed limit,	accelaration,	descelaration
0.004666666666666665,	0,	18,	0.6,	0.0056,	0.055
0.30062222222222224,	3.4671999999999983,	18,	0.6,	0.0056,	0.055
0.30062222222222222,	7.988399999999995,	18,	0.6,	0.0056,	0.055
0.30062222222222222,	11.188399999999996,	18,	0.6,	0.0056,	0.055
0.30062222222222222,	17.684199999999993,	18,	0.6,	0.0056,	0.055

- Gelijkwaardig kruispunt:  
Ondanks het feit dat het maar een verschil is van 6 auto's doet een gelijkwaardig kruispunt het wel al wat beter dan bij 6. We lijken er nog steeds vaak opstoppen te zijn in het verkeer.



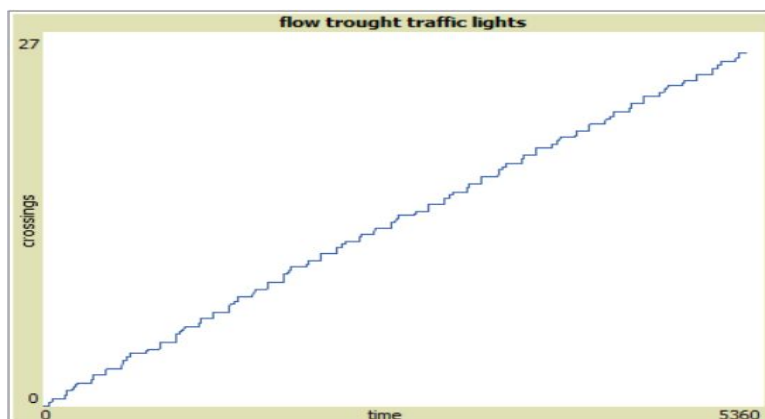
Ondanks dat er vaak opstoppen waren was de gemiddelde snelheid wel een stuk beter dan bij 6 auto's. De line count is amper gestegen en is nu ongeveer 2.5 in plaats van 2.

average speed,	line count,	total cars,	speed limit,	acceleration,	descelaration
0.00528888888888887,	0,	18,	0.6,	0.0056,	0.055
0.15301111111111104,	3.040799999999995,	18,	0.6,	0.0056,	0.055
0.1573333333333327,	4.972799999999999,	18,	0.6,	0.0056,	0.055
0.1794555555555555,	7.789599999999998,	18,	0.6,	0.0056,	0.055
0.1690444444444444,	9.654399999999997,	18,	0.6,	0.0056,	0.055

### 3 auto's:

#### - Stoplichten:

De stoplichten hebben een constant tempo en doorstroming net zoals bij 6 auto's. en bij 4-5 auto's. Wat je alleen wel ziet is dat auto's overbodig lang stilstaan. Zolang zelfs dat de andere auto's de tijd hebben om tweemaal over te steken. Wat erg onnodig is, want de helft van het verkeer moet dan wachten en dat verslechtert de doorstroming. Dit kan je alleen niet zien in de grafiek en in de staten van de simulatie.

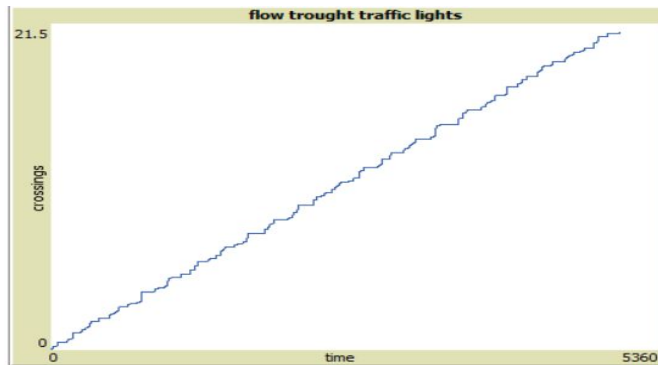


De average speed is bijna gelijk met dat bij 6 auto's en de line count is ongeveer 4-6.

average speed,	line count,	total cars,	speed limit,	acceleration,	descelaration
0.0056,	0,	12,	0.6,	0.0056,	0.055
0.3009333333333333,	3.2605999999999984,	12,	0.6,	0.0056,	0.055
0.3009333333333333,	9.782999999999992,	12,	0.6,	0.0056,	0.055
0.3009333333333333,	13.043599999999989,	12,	0.6,	0.0056,	0.055
0.3009333333333333,	19.565999999999995,	12,	0.6,	0.0056,	0.055

#### - Gelijkwaardig kruispunt:

Het kruispunt doet gaat ongeveer gelijk op met de stoplichten. Alleen hebben de stoplichten een onnodige lange wachttijd. Dit heeft het kruispunt niet, wat een groot voordeel is want er wordt nu constant doorgereden.



Wat we wel zien is dat de average speed veel hoger is dan bij 6 auto's en het is zelfs hoger dan bij de stoplichten. De line count is ongeveer 4-5 per stap wat ook een stijging is vergeleken met 6 auto's.

average speed,	line count,	total cars,	speed limit,	accelaration,	descelaration
0.0056,	0,	12,	0.6,	0.0056,	0.055
0.3069499999999995,	4.575999999999999,	12,	0.6,	0.0056,	0.055
0.33821666666666655,	9.655199999999997,	12,	0.6,	0.0056,	0.055
0.3003333333333333,	13.440799999999998,	12,	0.6,	0.0056,	0.055
0.31188333333333323,	17.383199999999995,	12,	0.6,	0.0056,	0.055

# Conclusie

## Onderzoeksvraag 1:

Een hogere maximumsnelheid zorgt voor een snellere oftewel hogere doorstroom. Echter zodra er een opstopping komt is het makkelijker voor een 50 weg om daar weer bovenop te komen. De 100 weg echter kan zelfs makkelijk onder de doorstroom van de 50 weg terecht komen. Na enige tijd wanneer de 100 weg de opstopping heeft opgelost wordt de doorstroom wel weer hoger dan die van de 50 weg.

## Onderzoeksvraag 2:

Als we kijken naar de resultaten kunnen we gelijk zeggen dat bij een hoge verkeersdichtheid de stoplichten een betere keuze zijn voor het verkeer.

Bij een gemiddelde verkeersdichtheid zien we dat een gelijkwaardig kruispunt het wel al beter begint te doen, maar de stoplichten weten het verkeer nog steeds soepel te besturen waardoor de doorstroming nog steeds beter blijft dan bij een gelijkwaardig kruispunt.

Als we kijken naar de resultaten bij een lage verkeersdichtheid zien we dat de stoplichten niet veranderen en de line count nog zelfs hoger is dan bij een gelijkwaardig kruispunt, maar een ding wat een groot verschil maakt voor het eindresultaat is dat (zoals voorheen gezegd) de wachttijd van de stoplichten te lang zijn wat ervoor zorgt dat de helft van het verkeer onnodig vaststaat. Een gelijkwaardig kruispunt heeft dit probleem niet en heeft bij een lagere verkeersdichtheid een net wat lagere line count, maar een hogere gemiddelde snelheid. Dus kunnen we zeggen dat bij een lage verkeersdichtheid een gelijkwaardig kruispunt de betere keuze is voor het verkeer.

Met dit onderzoek kunnen we ook gelijk de conclusie trekken dat de verkeersdichtheid zeker te maken heeft met welke oplossing beter is voor de doorstroming van het verkeer. Namelijk bij een hoge verkeersdichtheid een stoplicht, bij een gemiddelde verkeersdichtheid een stoplicht en bij een lage verkeersdichtheid een gelijkwaardig kruispunt.



# Discussie

## Bronnen

*NetLogo 6.1.1 User Manual*. (2019, 26 september). Geraadpleegd op 1 december 2020, van <https://ccl.northwestern.edu/netlogo/docs/>

Tisue, S. (2015, 15 augustus). *Exporting data to CSV file*. Geraadpleegd op 15 december 2020, van <https://stackoverflow.com/questions/32026692/exporting-data-to-csv-file>

*NetLogo Models Library: Traffic Grid*. (z.d.). Geraadpleegd op 2 december 2020, van <http://ccl.northwestern.edu/netlogo/models/TrafficGrid>

*NetLogo Models Library: Traffic Basic*. (z.d.). Geraadpleegd op 2 december 2020, van <http://ccl.northwestern.edu/netlogo/models/TrafficBasic>

Isaac, A. G. (z.d.). *NetLogo Programming: An Introduction*. Geraadpleegd op 15 december 2020, van <https://subversion.american.edu/aisaac/notes/netlogoProgramming.html>