**onderzoeksvragen:**

1. Wat is het effect van meerdere lanes opstellen op de verkeersdichtheid
2. Wat is het effect van een invoegstrook op de doorstroom
3. Wat is het effect van een vrachtwagen(s) op de doorstroom
4. Wat zal beter zijn voor de doorstroom. Stoplichten of een rotonde, en is bij een andere verkeersdichtheid de een misschien weer beter dan de ander.

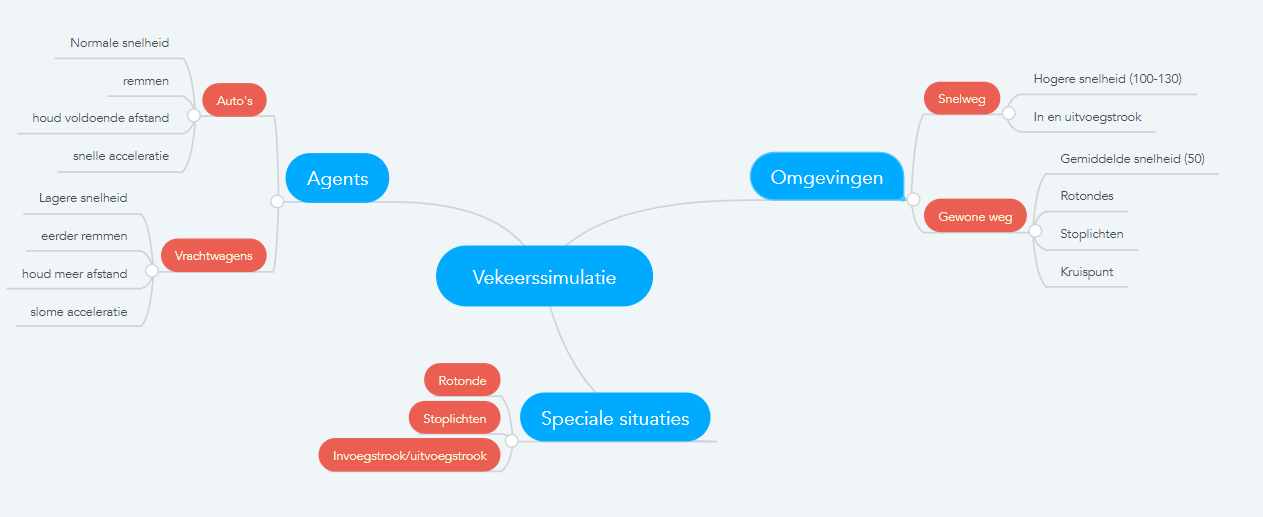
@ docent welke onderzoeksvraag is het beste (is goed) want dan kunnen we die hypothese uitbreiden.

**hypothese:**

1. Meerdere lanes zou waarschijnlijk zorgen voor een lagere verkeersdichtheid en een betere doorstroming aangezien de hoeveelheid auto’s op een baan dan verkleint wordt.
2. Waarschijnlijk wordt de doorstroom slechter, maar niet voor een lange periode.
3. Dit zal waarschijnlijk de doorstroom slechter maken. Aangezien vrachtwagens gemiddeld slomer moeten rijden dan normale auto’s.
4. Waarschijnlijk zal een stoplicht een betere doorstroom hebben bij een hogere verkeersdichtheid en een rotonde bij een lagere verkeersdichtheid

**Plan van aanpak:**

Voor onze simulatie hebben we een mindmap gemaakt:

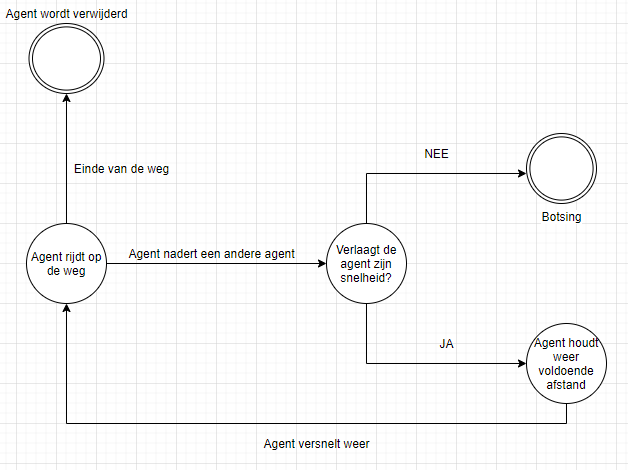


Als we verschillende dingen van deze mindmap combineren zouden wij de hoofdvraag en onze eigen onderzoeksvraag kunnen beantwoorden.

**Basis simulatie:**

De basis simulatie zal bestaan uit een simpele weg waar een max snelheid is. Auto’s kunnen niet sneller gaan dan de max snelheid. We zullen verdere de auto’s de waardes geven die in de mindmap staan en de hoeveelheid auto’s in de simulatie aanpasbaar maken om te kijken wat voor effect dit heeft op het resultaat.

De afbeelding hieronder is de basis FSM voor de agent (een auto):



**Eind simulatie:**

* **GUI:**

We zullen meerdere variabelen aanpasbaar maken door middel van bijvoorbeeld sliders (zoals de max snelheid, aantal auto’s en vrachtwagens en bepaalde specs van de agents) . Dit maakt het makkelijk om verschillende simulatie te runnen. Of we dit echt gaan doen door middel van een heftig GUI of door gewoon simpele inputs en sliders is nog de vraag.

* **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. Als er niet zoiets is als een tick of stap in de tool kunnen we werken met een timer. Die bijvoorbeeld om elke seconde alles runt.

Voor gebeurtenissen hebben we natuurlijk botsingen. Dit wordt niet random gemaakt. Er wordt eerder gekeken naar de afstand tussen 2 auto’s. Als deze heel klein is kunnen we dit zien als een botsing.

* **Staat:**

We kunnen kijken wat de staat van bepaalde waardes zijn. zoals de verkeersdichtheid (hoeveel auto’s op de weg), gemiddelde snelheid op de weg, aantal botsingen, en hoeveel auto’s bij een bepaald punt zijn gekomen. Dit kan bij bepaalde tools makkelijk weergeven worden met bijvoorbeeld monitors. Bij sommig 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 (zoals botsingen, tijd, enz.) en met behulp van dit kunnen we de resultaten over meerdere simulaties analyseren om onze onderzoeksvragen te beantwoorden.

* **Modules:**

Een auto en vrachtwagen hebben wij nodig om te kijken of vrachtwagens de doorstroom hinderen.   
Een snelweg hebben we nodig om te kijken of in/uit -voegstroken invloed hebben op de doorstroming van het verkeer op een snelweg. Ook hebben we de snelweg nodig om te testen wat de doorstroming doet op een snelweg 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.

**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, 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 |
| **Suitability** | **32** | **32** | **33** |
| 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 |
| **Feasibility** | **17** | **15** | **10** |
| Omgaan met tool? | 9 | 7 | 5 |
| Technisch haalbaar | 8 | 8 | 5 |
| **Totaal** | **49** | **47** | **43** |

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.