

Welk stelsysteem is het  
meest succesvol?

# Welk stelsysteem is het meest succesvol?



Gemaakt door: Koen van Heertum, Ruben van Raaij, Guy Veenhof

# Welk stelsysteem is het meest succesvol?

## Inhoud

|   |   |
|---|---|
| Onderzoeksvraag .....                     | 3 |
| Eerste inzicht.....                       | 3 |
| Setup .....                               | 3 |
| Go (voor nu favoriet plurality) .....     | 3 |
| Go (voor nu tweede keuze plurality) ..... | 3 |
| GUI .....                                 | 3 |
| Plan van aanpak .....                     | 4 |
| User stories .....                        | 4 |
| Planning .....                            | 6 |
| SF(A) model met modules voor tools .....  | 7 |
| Conclusie .....                           | 8 |

## Welk stelsysteem is het meest succesvol?

**Gekozen onderwerp:** Stelsystemen

**Eerder gebruikte onderzoeksvraag:** Wat is het effect van plurality voting op het aantal mensen dat niet op zijn eerste voorkeur kiest?

**Uitgebreide onderzoeksvraag:** Welke stelsysteem is het meest succesvol? (dat ook strategische stemmers op hun 1<sup>e</sup> keuze kunnen kiezen)

## Onderzoeksvraag

### Eerste inzicht

Om de uitgebreide onderzoeksvraag te beantwoorden, moet eerst de eerder gebruikte onderzoeksvraag beantwoord worden. Dit, in tegenstelling tot de uitgebreide, wordt er alleen gekeken naar plurality stelsysteem. Deze vraag kan beantwoord worden door de betrouwbaarheid te berekenen. De simulatie wat het verschil is tussen het strategisch kiezen (andere voorkeur, maar ten minste beter dan de slechtste) of favoriete partij (eerste voorkeur, maar grote kans dat de slechtste kan winnen). Om dit te simuleren moet de simulatie het volgende doen:

### Setup

- Cleared eerder gemaakte simulatie en maakt een nieuw wit vlak aan
- Maakt bepaald hoeveelheid kandidaten aan.
- 2d oppervlakte
- 4 Issues (gemaakte partij/kandidaat keuzes)
- Zet kandidaten bij willekeurige issues op de 2d oppervlakte
- Zet meer dan 2 (herkenbare)partijleiders op de oppervlakte en geef ze een (herkenbare) kleur

### Environment (voor nu favoriet plurality)

- Cleared alle kleuren van de kandidaten (behalve de partijleiders)
- Kandidaten zoeken naar de dichtbijzijnde partijleider
- Kandidaat veranderd naar de dichtbijzijnde partijleider's kleur
- Systeem telt welke kleur er het meeste van is
- De winnende partij(leider) krijgt een herkenbaar teken waarin hij gewonnen heeft

### Environment (voor nu tweede keuze plurality)

- Cleared alle kleuren van de kandidaten (behalve de partijleiders)
- Kandidaten zoeken naar de tweede dichtbijzijnde partijleider
- Kandidaat veranderd naar de tweede dichtbijzijnde partijleider's kleur
- Systeem telt welke kleur er het meeste van is
- De winnende partij(leider) krijgt een herkenbaar teken dat hij gewonnen heeft

### GUI

- Een model, wat de 2d oppervlakte weergeeft met daarin alle kandidaten en partijleider
- Een favoriet plurality histogram plot, wat de hoeveelheid getelde kleuren zijn op het model
- Een tweede keuze plurality histogram plot, wat de hoeveelheid getelde kleuren zijn op het model
- Een go knop, om de go functie (voor nu plurality) op aan te roepen

## Welk stelsysteem is het meest succesvol?

- Een setup knop, om de setup functie op aan te roepen
- Aantal kandidaten slider
- Aantal partijleiders slider

## Plan van aanpak

### Modules

- **GUI:** De grafische interface van het programma. Gaat over de GUI van het programma waarin we onze simulatie maken, en de GUI die inbegrepen is bij onze simulatie.
- **Agents:** De individuele 'nodes' van het programma. Deze vertonen allemaal individueel gedrag van elkaar, of werken allemaal een soort van samen. Het doel van de simulatie is om te kijken hoe deze agents reageren op hun omstandigheden/omgeving. Ze stemmen met behulp van een distance tot een partij(agent)
- **Environment:** De 'omgeving' van de agents. Deze kunnen we ook beïnvloeden, wat ervoor zorgt dat onze agents anders gaan reageren. Bijvoorbeeld door een policus op een andere plek in de omgeving te plaatsen, zullen de agents op een andere manier gaan stemmen.
- **Staat van simulatie toonbaar:** Hoe goed is het mogelijk om te bekijken wat er precies gebeurt met de agents? Is het goed mogelijk om een staatwisseling te zien?
- **Batchrun:** Ook wel batch mode. Hoe goed is het programma in het runnen van batch mode? Batch mode is het verwerken/compilen etc. van de code in de achtergrond, terwijl de computer op idle staat. Dit zodat de gebruiker geen last heeft van trage compiling zodra hij het programma wilt runnen.

### User stories

|                      |  |
|----------------------|--|
| <b>Titel:</b>        | GUI  |
| <b>Omschrijving:</b> | Als gebruiker wil ik een duidelijk overzicht waar ik ook de parameters kan veranderen zodat ik duidelijk kan zien voor welke parameters wat voor soort scenarios geven |
| <b>DoD:</b>          | Agents worden duidelijk weergegeven en er is een mogelijkheid geïmplementeerd om params te veranderen  |
| <b>Prio</b>          | 4 (1-5)  |
| <b>Tijdsduur:</b>    | 8 uur  |

|                      |   |
|----------------------|---|
| <b>Titel:</b>        | Agents  |
| <b>Omschrijving:</b> | Als agent wil ik kunnen kiezen welke partij het meest voldoet aan mijn standpunten (dit word gedaan met distance) en degene die het er daarna het meest aan voldoet zodat als elke agent dit doet je een duidelijk overzicht krijgt van hoeveel agents er voor elke partij heeft gekozen. |
| <b>DoD:</b>          | Als elke agent de mogelijkheid heeft om "toegewezen" te kunnen worden tot een partij  |
| <b>Prio</b>          | 5   |
| <b>Tijdsduur:</b>    | 2 uur   |

Welk stelsysteem is het  
meest succesvol?

|                      |  |
|----------------------|--|
| <b>Titel:</b>        | Environment  |
| <b>Omschrijving:</b> | Als gebruiker wil ik kunnen beïnvloeden in welke omgeving de agents geplaatst worden. Ik wil dus kunnen kiezen op welke plek mijn partijleiders terechtkomen en hoe de agents reageren op deze partijleiders. (op welke manier de agents zullen stemmen) |
| <b>DoD:</b>          | De agents moeten een omgeving hebben om neergezet te worden, en deze omgeving moet beïnvloedbaar zijn door de gebruiker.   |
| <b>Prio</b>          | 5  |
| <b>Tijdsduur:</b>    | 1 uur.   |

|                      |   |
|----------------------|---|
| <b>Titel:</b>        | Staat van simulatie toonbaar  |
| <b>Omschrijving:</b> | Als gebruiker wil ik kunnen zien wat de statussen zijn van agents binnen de environment zodat je kan zien wat de keuze/2 <sup>e</sup> keuze is van die agent. |
| <b>DoD:</b>          | Wanneer je op een agent clickt dan moet er informatie over die agent worden getoond   |
| <b>Prio</b>          | 3   |
| <b>Tijdsduur:</b>    | 10 min  |

|                      |   |
|----------------------|---|
| <b>Titel:</b>        | Batchrun  |
| <b>Omschrijving:</b> | Als gebruiker wil ik dat de simulatie meerdere keren kan runnen. Bovendien wil ik dat mijn computer dit op de achtergrond kan doen. |
| <b>DoD:</b>          | De simulatie moet meerdere keren te runnen zijn, eventueel met andere instellingen.   |
| <b>Prio</b>          | 2   |
| <b>Tijdsduur:</b>    | 30 min  |

|                      |  |
|----------------------|--|
| <b>Titel:</b>        | Setup  |
| <b>Omschrijving:</b> | Maakt aantal agents aan, agents eigenschappen, global variabele, plaats ze op oppervlakte en kiest verschillende agents als partyleader met kenmerkende eigenschappen. |
| <b>Prio</b>          | 4  |
| <b>Tijdsduur:</b>    | 2 uur  |

|                      |  |
|----------------------|--|
| <b>Titel:</b>        | Agent keuze 1  |
| <b>Omschrijving:</b> | Cleared eerst alle eerdere veranderde variabele en agents, roept de (vind de 1 <sup>e</sup> partij) functie aan en dan de functie countelections |
| <b>Prio</b>          | 4  |
| <b>Tijdsduur:</b>    | 3 uur  |

|                      |   |
|----------------------|---|
| <b>Titel:</b>        | Agent keuze 2   |
| <b>Omschrijving:</b> | Cleared eerst alle eerder veranderde variabele en agents, roept de (vind de 2 <sup>e</sup> partij) functie aan en dan de functie countelections |
| <b>Prio</b>          | 4   |
| <b>Tijdsduur:</b>    | 1 uur   |

|               |                               |
|---------------|-------------------------------|
| <b>Titel:</b> | Vind de 1 <sup>e</sup> partij |
|---------------|-------------------------------|

Welk stelsysteem is het  
meest succesvol?

|                      |  |
|----------------------|--|
| <b>Omschrijving:</b> | Elke agent gaat checken in zijn radius of er een partyleader is anders verbreed het zijn radius totdat het een partyleader vindt en dan zijn kleureigenschap veranderd naar de partyleaders en dan stopt de loop |
| <b>Prio</b>          | 4  |
| <b>Tijdsduur:</b>    | 5 uur  |

|                      |  |
|----------------------|--|
| <b>Titel:</b>        | Vind de 2 <sup>e</sup> partij  |
| <b>Omschrijving:</b> | Elke agent gaat checken in zijn radius of er een partyleader is anders verbreed het zijn radius totdat het een partyleader vindt en dan zijn kleureigenschap veranderd naar de partyleaders, dit doet hij 2 keer (om de 2de partyleader te vinden) |
| <b>Prio</b>          | 4  |
| <b>Tijdsduur:</b>    | 4 uur  |

|                      |   |
|----------------------|---|
| <b>Titel:</b>        | Clearboard (tijdelijk)  |
| <b>Omschrijving:</b> | Tijdelijke functie om alle keuzes van kandidaten te resetten en daarna deze functie eigenschappen/code regels in de go functies toepassen |
| <b>Prio</b>          | 4   |
| <b>Tijdsduur:</b>    | 1 uur   |

|                      |  |
|----------------------|--|
| <b>Titel:</b>        | CountElections   |
| <b>Omschrijving:</b> | Zet de globale tellingen variabelen op nul, telt per agent hoeveel er per kleur is en zet het in een variabele. De variabelen worden uitgeprint in de output |
| <b>Prio</b>          |  |
| <b>Tijdsduur:</b>    | 3 uur  |

|                      |   |
|----------------------|---|
| <b>Titel:</b>        | GUI uitbreiding   |
| <b>Omschrijving:</b> | Histogram van de zichzelf tellende code van het aantal kleuren en zet de in een plot met x-as de kleur code en y-as de hoeveelheid. En de output van CountElections waarin in exacte waarden te zien is welke kleuren meer votes hebben. En nog een klein notitieblokje om aan te geven welke output bij welke kleur hoort en de kleurcode nummer (white = 9.9) |
| <b>Prio</b>          | 4   |
| <b>Tijdsduur:</b>    | 2 uur   |

## Planning

### Week 1 planning

| Onderdeel | Beschrijving                               | Weekdeel | Persoon    |
|-----------|--|----------|------------|
| PvA       | User Stories opzetten                      | MA       | Iedereen   |
| PvA       | Onderzoeksvraag en eerst inzicht uitwerken | MA       | Iedereen   |
| PvA       | SF(A) modelkeuze met modules               | MA       | Iedereen   |
| Simulatie | Setup + GUI + Gofirst                      | MA       | Iedereen   |
| Simulatie | Gosecond + clearboard                      | MA-DI    | Ruben      |
| Simulatie | Findfirstparty                             | DI       | Guy & Koen |
| Simulatie | Findsecondparty                            | WE-DO    | Koen       |

Welk stelsysteem is het  
meest succesvol?

|           |                                  |       |          |
|-----------|----------------------------------|-------|----------|
| Simulatie | Countelections + GUI uitbreiding | DO    | Ruben    |
| PvA       | Laatste puntjes op de i          | DO-FR | Guy&Koen |
| Simulatie | Laatste puntjes op de i          | DO-FR | Ruben    |

### SF(A) model met modules voor tools

Scores zijn van 1-5, waarbij 1 het laagst is en 5 het hoogst haalbare is.

- Mesa: 29 punten
  - **Gui: 4**  
Je hebt een best wel moderne GUI en je kan ook het veranderen van parameters implementeren.
  - **Agents: 4**  
De agents zijn makkelijk op te zetten en makkelijk functies te maken voor de agents zelf.
  - **Environment: 4**  
De environment is ook makkelijk op te zetten en best simpel.
  - **Staat van simulatie toonbaar: 4**  
Voor extra informatie moet je het wel zelf inprogrammeren, dus het weergeeft niet vanuit zichzelf de status van een agent als je erop klikt/erover heen hoveret.
  - **Batchrun: 4**  
Je zou makkelijk een batchrun kunnen uitvoeren in Mesa. Het is al geïmplementeerd in Mesa.
  - **Suitability: 4**  
Qua efficiëntie is Mesa redelijk. Je moet nog steeds wel aardig wat code schrijven om iets simpels te maken. , Mesa heeft een aardige snelheid qua het runnen en displaying van de code, omdat Mesa in python word gebruikt is de compatibility hoog
  - **Feasability: 5**  
Iedereen weet hoe ze moeten coderen in python, dus er hoeft niet echt een nieuwe taal aangeleerd te worden. Mesa is aardig goed te begrijpen. Technisch is het zeker haalbaar om in 2 week iets goed in elkaar te zetten.
- Unity: 25 punten
  - **Gui: 5**  
De GUI in Unity is heel uitgebreid en zorgt ervoor dat alle nodige info zichtbaar is, mits je weet waar je het moet vinden.
  - **Agents: 4**  
Agents zijn goed op te zetten in Unity.
  - **Environment: 4**  
Een environment opzetten in Unity is vrij makkelijk.
  - **Staat van simulatie toonbaar: 3**  
Afhankelijk van de simulatie is dit makkelijk of moeilijk. Als je een visuele indicator toevoegt wordt het al een stuk leesbaarder, maar het is moeilijker dan in de andere 2 talen

## Welk stelsysteem is het meest succesvol?

- **Batchrun: 5**  
Batch is een zeer goed geïntegreerde command in Unity aangezien ze tijdens de development van Unity wisten dat compilen, runnen etc. lang zou kunnen duren.
- **Suitability: 2**  
Suitability is niet erg hoog. In Unity zou onze simulatie niet efficiënt zijn, de compilatietijd zou vrij lang duren. Waarschijnlijk zou Unity beter geschikt zijn voor de snelweg-simulatie.
- **Feasability: 2**  
Wij zijn alle 3 niet echt bekend met Unity/C#, en we zouden niet goed weten waar we zouden moeten beginnen. Technisch zou het waarschijnlijk wel haalbaar zijn, maar het zou langer duren dan nodig is en in andere talen is het vele malen makkelijker.
- Netlogo: 30 punten
  - **Gui: 5**  
De gui in Netlogo is al ingebouwd en staat als template al klaar om gebruikt te worden, het enige wat er gedaan moet worden is het aanmaken van turtles
  - **Agents: 4**  
Agents kunnen aangemaakt worden met 3 woorden, het is erg simpel om ze aan te maken en ze te inspecteren.
  - **Environment: 5**  
De basis van het environment wordt al aangemaakt als je een template aanmaakt en is daarom automatisch aangemaakt.
  - **Staat van simulatie toonbaar: 5**  
De agent informatie is erg makkelijk te zien door bij het environment op de agent te klikken en dan kan je het inspecteren en kan zien welke variabele met agent te maken hebben en welke waarde het heeft
  - **Batchrun: 3**  
Heeft BehaviorSpace i.p.v. batch mode. BS runt op een parallelle core op je CPU in plaats van wanneer je PC in idle mode is. BehaviorSpace werkt wel goed maar is dus in werkelijkheid geen echte batch mode.
  - **Suitability: 4**  
Heel efficiënt en snel, maar helaas niet compatible met externe libraries.
  - **Feasability: 4**  
We zijn allemaal in een korte tijd bekend geraakt met NetLogo, omdat het niet moeilijk is om te begrijpen. Hierdoor is ons project op een technisch niveau zeker haalbaar.

## Conclusie

Volgens het voting systeem wat wij boven hebben gebruikt. Is Unity de minst goeie oplossing voor een simulatie creëren. Netlogo is maar met een punt beter dan Mesa. Wij zullen dus Netlogo gaan gebruiken als tool.