

De invloed van strategisch stemmen

Gemaakt door:

Wail Abou: 1762891
Quinn de Groot: 1755585
Kai ter Horst: 1762818



Inhoudsopgave

inhoudsopgave	1
1. Abstract	2
2. Introductie	2
3. Onderzoeksvraag	3
4. Vooronderzoek	3
5. Toolkeuze	4
6. Design & uitleg experiment	4
7. Resultaten	5
8. Conclusie	7
10. Referenties	8

1. Abstract

Ons onderwerp van dit project is Instant-runoff bij een verkiezing. Bij deze opdracht konden wij kiezen tussen wat het effect is van de maximumsnelheid op de doorstroom van de weg, en wat het niet kiezen op de eerste keuze voor effect heeft bij verkiezingen. Hierbij hebben wij gekozen voor de verkiezing, omdat ons dit wel leuk leek en het de laatste tijd veel in het nieuws was.

Voor het maken van onze simulatie gebruiken we MESA. We hebben verschillende stem systemen geïmplementeerd en gevisualiseerd. Hierbij is instant runoff het belangrijkste want die hebben we nodig voor onze onderzoeksvraag. Per scenario veranderde we het aantal agents dat strategisch stemde. Naarmate het percentage dat strategisch stemde toe neemt nam zag je dat de partijen die normaliteit niet zouden winnen toch een kans hadden, dit komt omdat de grootte van de ruimte minder aan doet en de vorm van de ruimte belangrijker wordt. Specifiek een kruisvorm, dit komt omdat de voters de x of de y as als preference hebben waardoor ze als het ware in een kruis om die partijen heen vormen, dit heeft als effect dat de partij met de grootste kruis oppervlakte zal winnen i.p.v. de partij met het grootste lege ruimte om zich heen.

2. Introductie

Bij verkiezingen zie je vaak dat het begint met veel verschillende partijen, maar toch eindigt met maar 2 partijen, hoe kan dat? Een van de redenen hiervoor kan Instant-runoff zijn. Instant-runoff is het concept dat elke ronde een partij uit het proces wordt gehaald, en de stemmers van deze partij worden verdeeld over de andere partijen. Dit wordt herhaald totdat er uiteindelijk 1 partij de meerderheid van de stemmers heeft. Dit concept heeft veel voordelen en nadelen, maar wordt alsnog bij de meeste verkiezingen in de wereld gebruikt.

Wij gaan in dit project kijken naar een verkiezing met instant-runoff, en we gaan hierbij kijken wat het effect erop is als mensen niet kiezen voor de partij van hun eerste keuze, maar voor een andere partij kiezen die hun meest belangrijke punt handhaven.

Stemmen is een groot onderwerp en verschilt van land tot land, verder is het ook een zeer complex onderwerp waar psychologie een grote rol speelt. Door deze dingen en andere externe factoren is het moeilijk om een precieze simulatie te hebben, daarom gaan wij ook er voor om een versimpelde weergave te maken waarbij de kern van de stelsystemen duidelijk bij worden.

3. Onderzoeksvraag

Tijdens dit onderzoek willen we erachter komen wat voor effect heeft strategisch stemmen op de uitkomst van de instant-runoff stelsysteem? Met effect bedoelen we of de uitslag anders zal zijn wanneer er meer of minder mensen strategisch stemmen. Wij zijn vooral benieuwd naar de nieuwe winnende partij en waar die ligt ten opzichte van de vorige winnende partij.

Onderzoeksvraag: Wat voor effect heeft strategisch stemmen op de winnaar van de instant-runoff stelsysteem?

Hypothese: Wij denken dat een enkele strategische stemmer al verschil kan hebben op de uitslag.

normaliteit zou de partij die de meeste lege ruimte om zich heen heeft winnen, in praktijk kan je dit opvatten als een partij die missende standpunten vult waar mensen behoefte aan hebben. Stel voor je hebt een perfect 4 verdeling, dan zal ieder partij 25% van de stemmen krijgen bij Plurality voting, dit is een stem strategie waarbij een voter voor de dichtbijzijnde partij kiest oftewel een partij die het dichtst bij zijn mening komt.

Als 1 iemand dan strategisch zal stemmen zal 1 partij 1 stem meer hebben waardoor de hele uitslag veranderd, want als een iemand 1 meer heeft een ander 1 minder en zijn de overgebleven 2 niet veranderd wat inhoud dat er nu een 1ste, 2de en 3de plek is.

4. Vooronderzoek

Tijdens het vooronderzoek hebben we verschillende tools met elkaar vergeleken. Elk teamlid kreeg een tool toegewezen en moest daar de tutorial van maken. Hierdoor kwam hij achter de plus en minpunten van de verschillende tools. Mesa scoorde bij ons het hoogst. Mesa is namelijk een module voor python en dat is gelijk het eerste pluspunt, je hoeft namelijk geen nieuwe taal aan te leren. Ook kon het goed omgaan met verschillende datatypes en andere python modules zoals panda's. Unity was onze oorspronkelijke eerste keus maar niet elk groepslid had daar evenveel ervaring mee waardoor ons dat geen goede keus leek. Tot slot hebben we ook nog naar Netlogo gekeken maar dat leek ons te simpel waardoor we niet heel veel waarde uit deze opdracht zouden halen. Ook hebben we verschillende artikelen bekeken. We zagen weinig die precies hetzelfde als ons wilde doen maar dat zegt niet dat het niet bruikbaar was. Je zag bijvoorbeeld verschillende knelpunten voorbij komen waar ze tegenaan liepen en het is handig als je die van te voren weet zodat je er alert voor bent.

Toen we de tool keuze gemaakt hadden moesten we ook gaan denken over de opzet van de simulatie. Ons idee was om eerst een simulatie te maken die zowel de partijen als agents op een random plek spawn. Dit willen we uitbreiden door de partijen op een vaste plek te zetten. En de agents die stemmen een tactiek mee te geven. Het percentage dat strategisch stemt willen we per scenario aanpassen om te kijken wat de invloed daarvan is. De afbeelding die we gebruiken als referentie voor de locaties van de partijen heeft elf partijen, wij schrappen er eentje zodat er meer plek is in het midden.

5. Toolkeuze

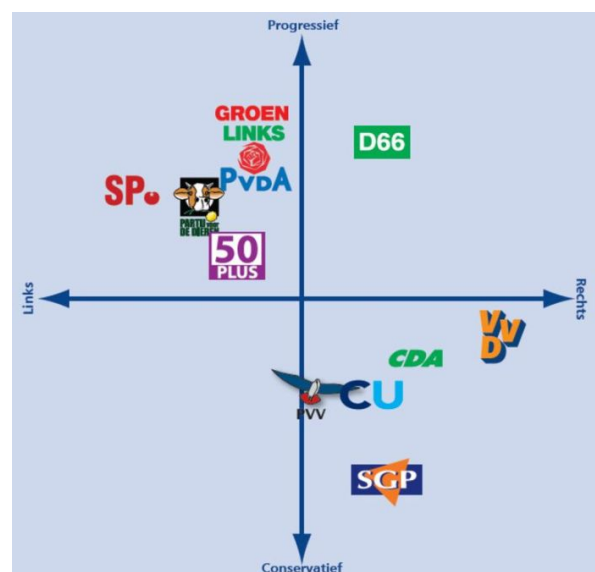
We keken naar verschillende modules om onze tool keuzes te bepalen. De eerste is agent spawning, deze module gaat over het aanmaken van agents en hoe lastig is, ook hebben we gekeken naar wanneer je ze kan toevoegen aan de simulatie. De tweede module is partijen aanmaken. De partijen zijn de kandidaten waarop de agents kunnen stemmen. Ook hier hebben we gekeken hoe lastig het is om die aan te maken en verschillende eigenschappen mee te geven. Als derde module hadden we, veranderen van de stem strategie van de agent. Hier keken we of het mogelijk is een agent een eigen strategie te geven zodat je een mix van strategieën kan maken per simulatie. Ook willen we de winnende partij kunnen visualiseren en dat is onze vierde module. We keken dus op welke manier we de uitkomst kunnen visualiseren en of die manier handig is als je het eindresultaat wilt analyseren. Visualiseren op welke partij een agent gestemd heeft is de vijfde module. We wilde dus weten of het mogelijk is om bij te houden op welke partij een agent gestemd heeft. De laatste module is data opslaan. Hier keken we hoe de tool data opslaat en met welke data hij om kan gaan zodat wij er nuttige dingen mee kunnen doen. Onze uiteindelijke keuze is MESA geworden, die kan namelijk met elke module die we hadden goed overweg en brengt weinig beperkingen met zich mee.

6. Design & uitleg experiment

We laten elk scenario duizend keer runnen en kijken dan hoe vaak een partij gemiddeld wint en of er dus een duidelijk verschil is tussen de scenario's. Ook onderzoeken we dus gelijk of er een duidelijk verschil is wanneer er een kleine groep strategisch stemt of een grote groep. We gebruiken elke keer 1000 agents zodat de matrix niet te vol zit en daar ook nog enige randomness is zit. Tijdens dit experiment gebruiken we 10 partijen waarop de agents kunnen stemmen. Dit hebben we gedaan zodat we de afbeelding hieronder kunnen na maken. De positie van de partijen veranderen we dus niet tijdens de verschillende scenario's. Wat we wel veranderen is het gedrag van de agents die stemmen.

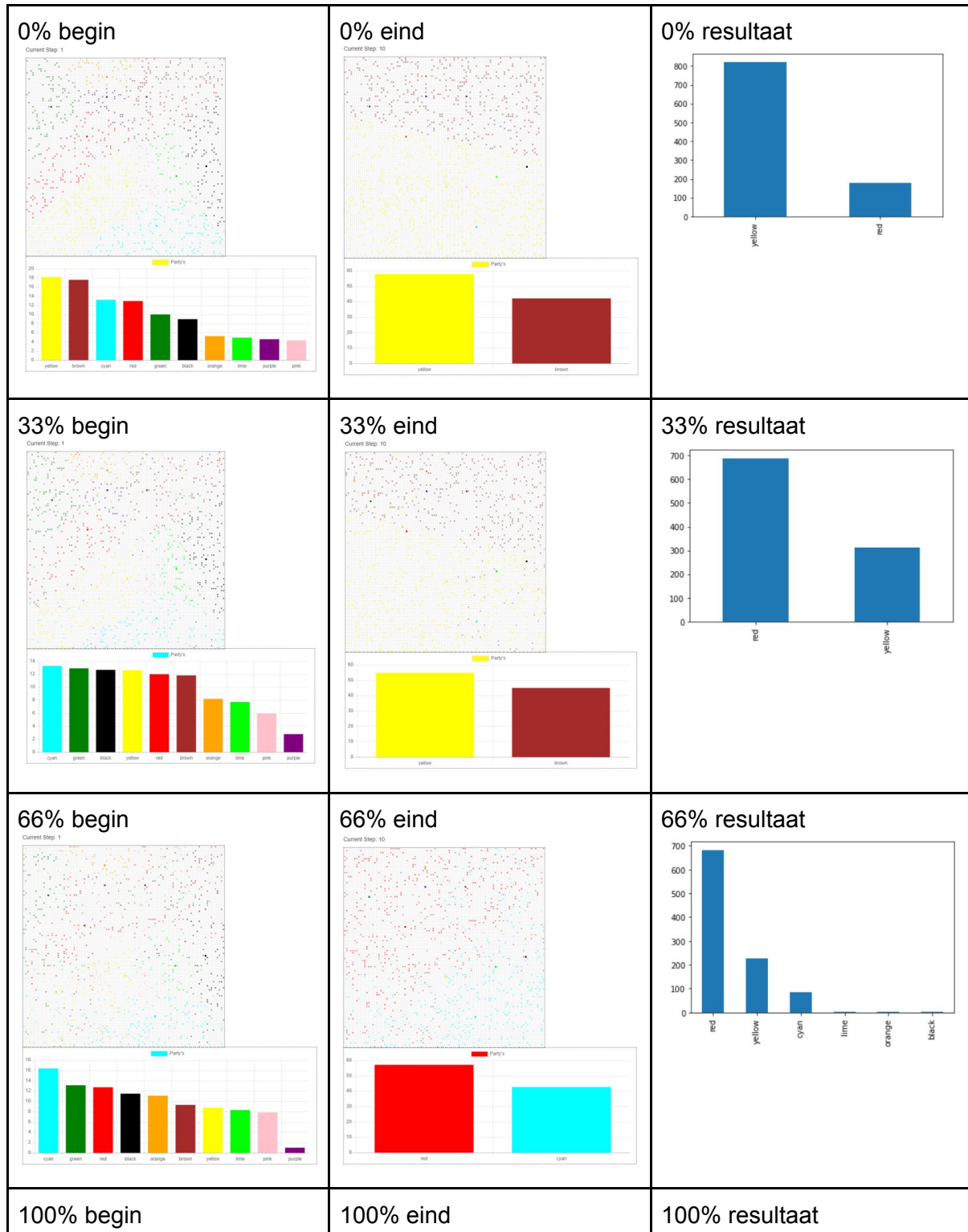
Een agent die normaal vote kijkt bij welke partij het dichtbij ligt. Dit berekenen we door te kijken naar de X en de Y as te kijken en vervolgens de Manhattan distance er van te pakken zodat we als het ware een rechte lijn meten naar de dichtstbijzijnde partij.

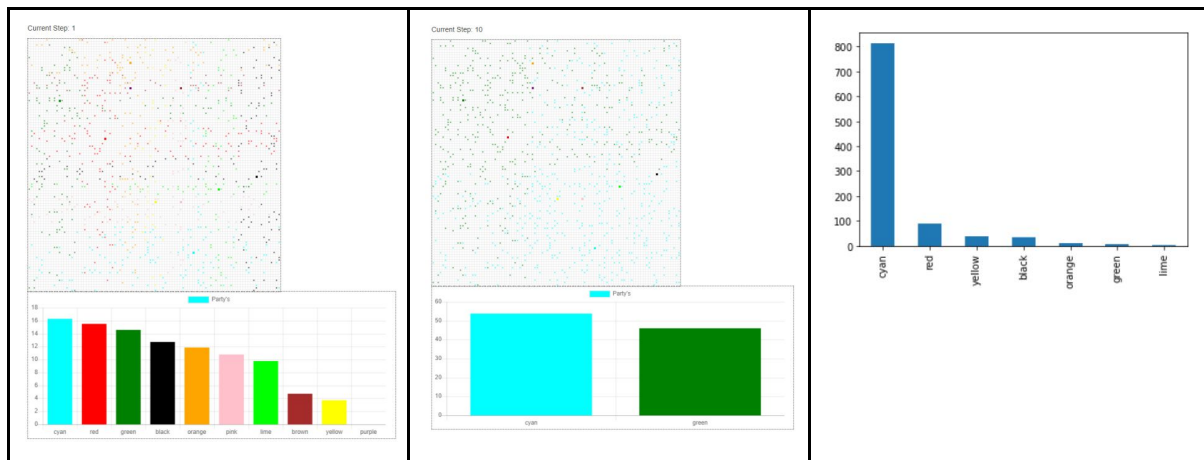
Agents die met tactiek stemmen kijken alleen naar de X of Y as, op deze manier wilde we simuleren dat een stemmer maar een punt belangrijk vindt. Tijdens de null meten geven we geen tactiek mee agents. Vervolgens laten we de agents wel strategisch stemmen, eerst 33% want dat is het aantal dat in de werkelijkheid strategisch stemt. Daarna 66% en tot slot 100%. Op deze manier komen er dus nog 3 verschillende scenario's bij. 33% leek ons een logisch beginpunt en dan kom je vanzelf aan stappen van 33%.



7. Resultaten

Resultaten de verschillende scenario's. De linkse rij geeft een beeld weer de verdeling in de eerste ronde. De middelste laat de verdeling zien in de laatste ronde. De rechtse rij laat zien hoe vaak een partij gewonnen heeft in de duizend keer dat het scenario gerund is.





We kunnen zien naarmate de strategische stem strategie toeneemt er meer een grid ontstaat, als we dit naar het echte leven vertalen zijn de meningen meer verspreid. In het begin zien we een duidelijke doorgetrokken lijn maar naarmate de strategisch stemmen percentage toeneemt lopen die lijnen meer door elkaar heen.

We zien dat bij de lage percentages de partijen met het meeste 'bedekte' oppervlakte winnen. We bedoelen met oppervlakte de lege ruimte, oftewel hoe het naar het echte lege zou vertaald worden: de mensen met een mening die niet/slecht worden gerepresenteerd door de partijen. Door dit zullen deze voters proberen de dichtste alsnog te vinden ook al zit die partij redelijk ver.

Houd in gedachten dat volgens onderzoek(zie de papers) grofweg 33% strategisch stemt. Met dit in gedachten lijkt ons instant runoff een beter strategie dan plurality omdat we zien dat bij deze percentage de winnende partij drastisch kan veranderen vergeleken met Plurality voting (eigenlijk hetzelfde als instant runoff met 0% strategie zie bovenstaande tabel).

8. Conclusie

Tijdens dit onderzoek wilden we erachter komen wat voor effect heeft strategisch stemmen op de winnaar van de instant-runoff stelsysteem?

Onze hypothese hierbij was: Wij denken dat een enkele strategische stemmer al verschil kan hebben op de uitslag.

We hebben alle scenario's 1000 keer laten runnen met 1000 voters en 10 partijen, zodat we een duidelijk gemiddelde als resultaat krijgen. Zoals je ziet wint partij geel het vaakst wanneer er geen strategie is. Naarmate de hoeveelheid agents die strategisch stemt toeneemt zie je dat rood vaker gaat winnen dan geel, en dat cyaan in de laatste scenario de meeste keer gewonnen heeft.

Bij 0% zie je dat geel het meest wint. Dit komt omdat bij 0% geel en rood het grootste 'bedekte' oppervlakte hebben, die van geel is echter net iets groter, waardoor deze iets vaker wint.

Bij 100% zie je dat cyaan de meeste overwinningen heeft. Dit komt omdat bij 100% een kruis formatie wordt aangenomen, en hierbij hebben andere partijen een kans om te winnen, en cyaan heeft de grootste kruis formatie waardoor deze partij het meest wint.

Bij 33% en 66% zie je dat het een overloop is tussen de twee formaties, waarbij ze elke keer 1 gedragpunt overnemen. Zoals bij 33% wordt het gedragpunt overgenomen dat rood vaker wint dan geel, en bij 66% wordt het gedragpunt dat meer dan 2 partijen een overwinningen kunnen hebben overgenomen.

Onze hypothese is dus correct. Dit is vooral te zien bij de overgang van 0% naar 33%, waarbij rood meer strategische stemmers krijgt dan geel, en daardoor meer overwinningen behaalt. Bij 1000 voters is het misschien niet 1 strategische voter die het verschil maakt, maar bij 100 voters zou dit misschien wel het geval kunnen zijn.

9. Discussie

We hebben de scenario's gerund met duizend agents per keer. Zodat de grid niet helemaal vol zit en in de positie ook nog randomness in zit. Ons leek het dat dit geen invloed zou moeten hebben op de uitslag omdat de kans even groot blijft dat een agent in dat gebied spawnst en de verdeling dus grofweg hetzelfde blijft en dus niet afhankelijk is van hoeveel agents er zijn. Een vervolgonderzoek zou dan ook kunnen zijn om de simulatie per percentage interval te runnen met verschillende aantal voters.

Wat we verwachtte was dat partijen die dicht bij het midden zaten de grootste kans hebben om te winnen maar met de tactiek die wij geïmplementeerd hadden bleek dit niet het geval. Ons leek dit logisch omdat deze partijen in de realiteit proberen overeen te komen met de belangen van de gemiddelde stemmer en daardoor meer stemmen proberen te verkrijgen in tegendeel tot wanneer je extreem rechts of links bent. Bij onze simulatie lag het vooral aan de plek die partijen van elkaar verwijderd zaten omdat die partij dan voor veel agents de dichtstbijzijnde partij is.

De resultaten die wij uit deze simulatie hebben gewonnen zijn lastig over te brengen naar de werkelijkheid. Waar iemand op stemt hangt normaal af van veel meer factoren die wij buitenwegen hebben gelaten. Dit is dus eigenlijk ook gelijk het zwakke punt van onze simulatie. We denken achteraf dat het eigenlijk te simpel is om een accuraat beeld te schetsen over de werkelijkheid maar het is wel een goed begin waar makkelijk op uit te breiden is.

Het is bijvoorbeeld uit te breiden door meerdere variabelen toe te voegen die stemmers belangrijk vinden zoals algemene standpunten die partijen hebben of lijsttrekkers die agenten wel of niet mogen. Door dit te doen krijg je ook gelijk een beter beeld van de partijen. Die kunnen al hun standpunten tonen zodat de agents die kunnen vergelijken met wat ze zelf belangrijk vinden.

10. Referenties

Edward Fieldhouse, Nick Shryane, Andrew Pickles, Strategic voting and constituency context: Modelling party preference and vote in multiparty elections, Political Geography, Volume 26, Issue 2, 2007, Pages 159-178, ISSN 0962-6298, <https://doi.org/10.1016/j.polgeo.2006.09.005>.

Singh V.K., Basak S., Modanwal N. (2011) Agent Based Modeling of Individual Voting Preferences with Social Influence. In: Nagamalai D., Renault E., Dhanuskodi M. (eds) Trends in Computer Science, Engineering and Information Technology. CCSEIT 2011. Communications in Computer and Information Science, vol 204. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-24043-0_55

Mesa Overview — Mesa .1 documentation. (2016). Mesa. <https://mesa.readthedocs.io/en/stable/overview.html>