Adviesrapport RL

verkeerslichten in de stad

Esmeralda de Wolf & Stan Meyberg
04-06-2021

Situatieschets

Om een beter beeld te krijgen van het probleem dat uiteindelijk opgelost moet worden worden eerst kort bestaande oplossingen toegelicht. Met behulp van de bestaande oplossingen kunnen de voor en nadelen van RL ten opzichte van deze oplossingen aangekaart worden. Hierbij zal per voor en nadeel ook toegelicht worden wanneer deze optreedt.

Bestaande oplossingen

Fixed Time Control

Bij Fixed Time Control wordt gebruik gemaakt van een vooraf ingestelde cyclus en faseplan. Hierbij blijft een verkeerslicht dan voor bijvoorbeeld 60 seconden op rood staan om daarna 20 seconden op groen te gaan. Deze methode wordt voornamelijk toegepast wanneer de verkeersstroom stabiel is (Wikipedia, 2021; Wei et al., 2018).

Coordinated Control

Bij Coordinated Control worden de clusters gecontroleerd door een master controller. Hierbij worden de verkeerslichten bij meerdere kruispunten op elkaar afgestemd dat kolonnes van voertuigen door een continue reeks van groene lichten kunnen rijden. Deze lichten worden dan dusdanig op elkaar afgestemd dat voertuigen, zonder te stoppen, door kunnen rijden zolang de snelheid van het voertuig lager ligt dan de gegeven limiet. Dit systeem wordt ook wel de “groene golf” genoemd (Wikipedia, 2021).

Self-Organizing Traffic Light Control

Bij Self-Organizing Traffic Light Control worden de verkeerslichten gebaseerd op de huidige verkeerstoestand. Er wordt gekeken naar hoeveel tijd er is verstreken sinds het licht op rood staat en hoeveel auto’s er staan te wachten. Specifiek verandert het verkeerslicht wanneer het aantal wachtende auto’s boven een met de hand afgestemde drempelwaarde ligt (Wei et al., 2018).

Smart Traffic

Bij Smart Traffic wordt data van camera’s, lussen in de weg en GPS signalen gecombineerd en worden deze geanalyseerd. Vervolgens wordt dit in een realtime model gebruikt om zo het verkeer te voorspellen (Sweco Nederland, z.d.). Smart Traffic berekent de meest optimale doorstroming en past de aansturing van de verkeerslichten hier op aan. Sinds december 2020 zijn er in Nederland al meer dan 700 verkeerslichten die gebruik maken van dit model (Steinbuch, 2021).

Voordelen RL

- RL methoden kunnen direct leren van geobserveerde data zonder onrealistische assumpties (zoals dat de verkeersstroom uniform is voor een bepaalde periode) te

maken, door eerst acties te nemen en vervolgens te leren van de uitkomsten (Wei et al., 2020).

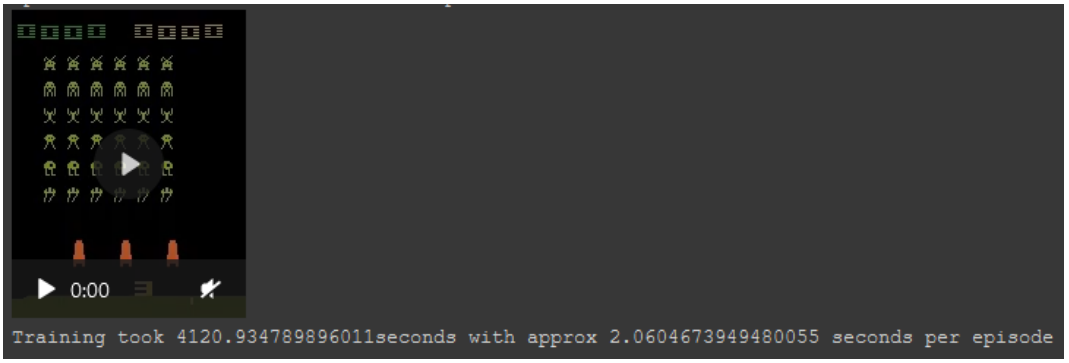
- Conventionele oplossingen worden ingesteld op bijv. hoe druk het is tussen bepaalde tijden. Echter kan deze drukte over tijd veranderen waardoor de huidige instellingen van de verkeerslichten niet meer kloppen voor het daadwerkelijke verkeer. Hier hebben RL methoden veel minder last van. Dit komt omdat RL de huidige staat van het verkeer (zoals aantal wachtende auto's, geupdate wachttijd) gebruikt om zo te proberen acties te nemen die de staat van het huidige verkeer verbeteren (Wei et al., 2021).
- In de huidige toepassingen moeten ofwel alle cases uitgewerkt worden ofwel veel berekeningen gedaan worden waarbij ook rekening gehouden moet worden met omliggende clusters. Het voordeel van een RL methode is dat wanneer de omgeving en de acties gedefinieerd zijn de agent zelf uitzoekt wat de beste waarde is voor iedere staat. In de CartPole oefening was dit dan ook goed te zien[1]. Nadat de huidige staat gediscretiseerd was kon de agent, na de training, op basis van de staat en de mogelijke acties de beste actie kiezen.

Nadelen RL

- Bij Fixed Time Control zit er een vooraf bepaalde tijdsinterval tussen voordat het verkeerslicht op groen zal springen en bij Coordinated Control kan door deductie beredeneert worden wanneer een verkeerslicht op groen zal springen. Met een RL model is het vrij moeilijk te voorspellen wat de volgende actie van het model zal zijn. Bij het gebruik van een RL model voor het aansturen van verkeerslichten wordt er controle verloren over de aansturing (Wei et al., 2020).
- De agent en de omgeving interacteren met elkaar. Iedere tijdstap maakt de agent een actie volgens een bepaald beleid en krijgt hiervoor een bepaalde reward. Het uiteindelijke doel van de agent is om zijn beleid zo te verbeteren dat de agent uiteindelijk de maximale reward kan krijgen (The TF-Agents Authors, 2021). Echter, de agent heeft geen context. De agent zoekt dus naar de beste reward en kan hierbij dus stappen nemen die niet gewild of op het randje zijn wat betreft veiligheid en traffic flow.
- RL methodes maken gebruik van trial-and-error. Deze manier van leren kan kritiek of zelfs fataal zijn in de echte wereld, aangezien het slecht functioneren van verkeerssignalen tot ongelukken kan leiden (Wei et al., 2020, 16). Door deze manier van leren is het dan ook vrijwel onmogelijk om de agent te trainen in de echte wereld. Om deze reden moet er een simulatie aan te pas komen. Alhoewel een simulator verschillende verkeerssituaties kan simuleren zal het nooit in staat zijn om dit 100% accuraat te doen. Uiteindelijk zal het model dus gevalideerd moeten worden in de echte wereld wat qua veiligheid niet ideaal is (Schneider, 2020).
- RL modellen hebben over het algemeen veel parameters om te zetten voordat er getraind wordt. Het fine-tunen van deze parameters kan best een uitdaging vormen. Tegelijkertijd moet, door de manier van trainen, de agent veel iteraties over het model doen voordat hij überhaupt op een punt komt dat hij stappen in de gewenste richting kan maken. Zo kwam tijdens de experimenten naar voren dat het model, die gebruik

[1] https://github.com/StanMey/Adaptive_systems/blob/main/RLcolab/1_Hogeschool_Utrecht_Reinforcement_Learning_project_Q_Learning.ipynb

maakt van deep-learning om een atari game te spelen, ongeveer een uur en een kwartier bezig is met het draaien van 2000 iteraties.



- Het aansturen van, meerdere clusters, verkeerslichten om de doorstroming te verbeteren is een complexe taak waarbij dan ook een complex model nodig is. Deze complexiteit zou ervoor kunnen zorgen dat het model niet op de goede manier traint of presteert.

Ethiek

De eerste vraag die naar voren komt als het gaat over ethiek en het optimaliseren van de traffic flow is, of het wel gewild is om de traffic flow te optimaliseren aangezien deze optimalisatie als een incentive zou kunnen werken voor mensen om sneller de auto te pakken. Volgens het principe van induced demand kan het elimineren van opstoppingen ervoor zorgen dat juist meer mensen hun gewoonte aanpassen en nu ineens de auto pakken (Speck, 2018, 64-65). Tegelijkertijd kan dit probleem ook bij de gemeente Utrecht gelegd worden. De keuze om de doorstroom te verbeteren komt uiteindelijk hiervandaan en het is om die reden dan ook logisch dat de gemeente voorafgaand van dit project de kosten tegen de baten zal afzetten om op basis van de afweging de beste richting te kiezen.

Een ander punt wat betreft ethiek behelst veiligheid. Dit kan opgedeeld worden in twee delen: menselijk gedrag en train gedrag. Wat betreft menselijk gedrag komt het punt naar voren dat mensen lak aan regels kunnen hebben (door bijvoorbeeld door rood te rijden). Dit soort gedrag valt helaas niet te voorkomen. De mens heeft helaas de neiging om zo nu en dan een eigen interpretatie te hebben van de verkeersregels.

Wat betreft het train gedrag van de agent zou de manier waarop de agent getraind wordt de veiligheid in het gedrang kunnen brengen. Dit omdat een agent probeert de beste reward te behalen waardoor hij misschien acties zal uitvoeren die negatief zijn voor de veiligheid. Dit zou wellicht voorkomen kunnen worden door de agent rekening te laten houden met bepaalde metrics of door de acties die genomen kunnen worden af te bakenen. Tijdens de training zou de reward functie zo ingesteld kunnen worden dat de agent gestraft wordt met een lage reward iedere keer dat er een aanrijding/noodstop is. Hierdoor zal een agent gestimuleerd worden om, naast de doorstroom te bevorderen, ook op de veiligheid te letten om zo een zo hoog mogelijke reward te kunnen behalen (Schneider, 2020).

Er zijn echter geen studies gevonden waarbij expliciet onderzoek is gedaan naar de impact van agent op de verkeersveiligheid. Sommige auteurs hebben grens condities toegevoegd aan hun agent om veiligheid te kunnen garanderen (bijv. een fase geel-rood) maar geen van de studies hebben dit expliciet toegevoegd aan het doel van de agent (Schneider, 2020).

Esmeralda de Wolf & Stan Meyberg
04-06-2021

Uitvoerbaarheid

Verkrijgen en verwerken van data

Uiteindelijk is het het idee om reinforcement learning te gebruiken om één of meerdere clusters van verkeerslichten in de stad aan te sturen. Aangezien RL technieken leren via een trial-and-error methode, en dit om veiligheidsredenen dus niet in het echt kan plaatsvinden, is er een behoefte voor het gebruik van een simulatie. Wat betreft de simulatie zelf zijn bijvoorbeeld CityFlow[2] of de software van Anylogic[3] goede kandidaten. Desalniettemin zou het voor het trainen ook interessant zijn om beschikking te hebben over data wat betreft de verkeersintensiteit op op bepaalde dagen en tijden voor ieder kruispunt.

In de praktijk komt er ook redelijk wat data binnen. Hieronder vallen de huidige staat waarin de verkeerslichten zich bevinden, maar ook informatie over waar auto's staan (via de lussen onder het wegoppervlak). Hierbij zou zelfs nog de hoeveelheid auto's die zich in de richting van het kruispunt bewegen vanaf een bepaalde richting zoals ook in het SURTRAC systeem meegenomen wordt (Smith et al., 2013, 436). Deze data wordt beschikbaar gesteld op twee manieren. Ofwel de controller die de verkeerslichten aanstuurt heeft al weet van de huidige staat. Ofwel er zijn sensoren (in het wegoppervlak of langs de weg) die informatie doorsturen.

RL in de praktijk

In de praktijk wordt de optimalisatie van de traffic-flow veelal aangepakt door middel van lokale optimalisatie met behulp van adaptive verkeerslichten. Ook wordt de groene golf methodologie gebruikt om meerdere clusters samen met elkaar te werken. Een mooi voorbeeld van een real-time adaptive system is het SURTRAC systeem. Dit systeem maakt voor iedere cluster een planning en communiceert de traffic flow ook direct naar de omliggende clusters die het mee nemen in de planning.

Als voor het optimaliseren van de traffic-flow gebruik gemaakt zal worden van een RL methode kan lering getrokken worden uit een paper (Liu et al., 2017) waarin verkeerslichten aangestuurd worden met behulp van multi-agent Q learning.

Om de veiligheid te garanderen wordt er namelijk gebruik gemaakt van discrete acties en worden er constraints en regels toegevoegd aan het algoritme dat verantwoordelijk is voor het leren. De discrete acties die gedaan kunnen worden zijn een set van alle mogelijke (veilige) acties die gedaan kunnen worden. Door het toevoegen van deze limitaties zal het model dan ook beter (en veiliger) toegepast kunnen worden in de echte wereld. Ook kan de agent ingesteld worden dat wanneer bijvoorbeeld voetgangers oversteken er voor een bepaalde tijdspanne geen acties genomen kunnen worden.

[2] https://cityflow-project.github.io/
[3] https://www.anylogic.com/road-traffic/

Om de agent te leren dat de optimalisatie van de traffic flow hoge prioriteit heeft is het dan ook gewild om de reward functie hierop af te stellen. In de praktijk kan de agent een negatieve reward gegeven worden op basis van de lengte van de wachtrij van auto's. Ook zou bijvoorbeeld de wachttijd gebruikt kunnen worden als negatieve reward. Aangezien de agent getraind wordt op ofwel de wachttijd ofwel de lengte van de wachtrij ofwel beide zal het zich bij verschillende verkeersdruktes ongeveer hetzelfde gaan gedragen. De focus zal namelijk liggen op het minimaliseren van de metrics wat een hogere reward oplevert.

Uiteindelijk moet het model nog wel steeds getraind worden met behulp van een simulator. Hierdoor blijft de kans bestaan dat het de overdraagbaarheid van het model naar de 'echte' wereld niet ten goede komt.

Planning

Wat betreft de planning zijn er vijf fases geïdentificeerd:

1. Opzetten van programma van eisen (PvE)
2. Uitvoeren metingen voor overzicht verkeerssituatie
3. Uitwerken simulatie
4. Opzetten en trainen model
5. Validatie van model

Deze vijf fases zullen nu allemaal kort toegelicht worden. Hierbij worden de taken, de tijdsplanning, de mogelijke risico's en de mitigatie van de risico's toegelicht.

Opzetten programma van eisen

Nog voordat begonnen kan worden met het uitwerken van een model of het uitvoeren van metingen moet een programma van eisen in elkaar gezet worden. Een programma van eisen helpt om van tevoren de randvoorwaarden en limieten te definiëren. In deze PvE wordt onder meer aangegeven wat de huidige situatie is, over hoeveel clusters het precies gaat, hoe groot de clusters zijn, welke maatregelen qua veiligheid genomen moeten worden, welke risico's komen voor en hoe deze te mitigeren, hoe de oplossing geëvalueerd moet worden, etc. Ook forceert een PvE het dialoog met de stakeholders, zoals bijvoorbeeld met Sharon Dijksma of met ervaringsdeskundigen. Uiteindelijk zorgt het PvE er dus voor dat alle facetten van het project doorgesproken worden zodat eventuele verwarringen weggefilterd worden.

Dit is een erg belangrijk onderdeel en daarom wordt ingeschat minstens 2-3 weken nodig te hebben hiervoor.

Overzicht verkeerssituatie

Er zijn verschillende manieren om de verkeerssituatie te meten. Er zou gebruik gemaakt kunnen worden van camera's die dicht in de buurt van kruispunten staan en iedere seconde een nieuwe data record aanmaken met daarin de cameraID, tijd en informatie over voertuigen (Wei et al., 2018). Verder zouden de GPS signalen van de auto's verzameld kunnen worden. Ook zijn er sensoren in de weg (lussen) die detecteren wanneer er auto's staan te wachten voor een verkeerslicht. Hiervan kan ook data verzameld worden (Sweco Nederland, z.d.). Tot slot zou er data van Google aangevraagd kunnen worden aangezien deze samen met het TNO geanonimiseerde locatiegegevens verzameld van mensen met Google-apps op de telefoon (inclusief IPhone gebruikers). Hier vraagt Google geen geld voor (Noort, 2015).

Er moet ook gekeken worden naar over welke tijd de data verzameld zal worden. In de paper 'IntelliLight: A Reinforcement Learning Approach for Intelligent Traffic Light Control' werd er data over 1 maand verzameld. Dit zou eventueel als richtlijn gebruikt kunnen worden. De inschatting wat betreft benodigde tijd voor dit onderdeel is ongeveer anderhalf tot twee maanden.

Uitwerken simulatie

Voordat er een model opgezet en getraind kan worden moet eerst de simulatie zelf worden uitgewerkt. Er moet gekeken worden naar de scope van de simulatie dus bijvoorbeeld welke kruispunt(en) zullen gesimuleerd worden. Er kan ook de keuze worden gemaakt om meerdere simulaties uit te werken voor ieder cluster van kruispunten.

Ook moet er gekeken worden naar welke factoren precies zullen worden meegenomen in de simulatie. De echte wereld bevat erg veel factoren en er zullen dus keuzes worden gemaakt over welke relevant en belangrijk zijn voor de simulatie. Zullen er bijv., naast auto's, ook ander verkeer zoals voetgangers, fietsers etc. worden gesimuleerd? Over hoeveel tijd zal er worden gesimuleerd (24 uur, 1 week, 1 maand)? Tot slot moet er ook gekeken worden welke software er zal worden gebruikt voor het uitvoeren van de simulatie. Een paar voorbeelden die eventueel gebruikt zouden kunnen worden: CityFlow[4], SUMO[5], Mesa[6]. Eventuele risico's die hier voor kunnen komen is dat het niet zou kunnen lukken om de simulatie werkend te krijgen. Wanneer dit het geval is zou verder gekeken moeten worden naar een model die wel naar behoren zal werken.

De inschatting voor de tijd die benodigd is voor dit onderdeel is 1 maand. Het uitwerken van de simulatie zelf is namelijk een vrij belangrijk onderdeel en daarom is het verstandig daar ruim de tijd voor te nemen.

Opzetten en trainen model

Voor het opzetten en trainen van het model moeten er een aantal keuzes gemaakt worden. Allereerst moet er een keuze gemaakt worden over welk RL model/algoritme gebruikt zal worden. Daarnaast moet de reward functie ingesteld worden.

Verder heeft ieder RL model zijn eigen hyperparameters die ingesteld moeten worden. Hiervoor zouden standaard waarden kunnen worden gebruikt als startpunt tijdens het trainen. Er zijn bij ieder RL model veel hyperparameters betrokken en iedere combinatie van waardes kan de uitkomsten van het trainen beïnvloeden. Verder duurt het, zoals al eerder benoemd, vaak veel erg lang om 1 model te trainen. Het kan verstandig zijn om in dit geval meerdere modellen tegelijkertijd te trainen met ieder net andere waardes voor de hyperparameters.

Voor dit onderdeel wordt ingeschat minstens 1 maand nodig te hebben, aangezien het trainen veel tijd kost en het instellen van de juiste hyperparameters ook vrij lastig is. Dit is dan ook direct waar de risico's van deze stap vandaan komen. Het trainen van het model kan langer duren dan vooraf ingecalculeerd wordt waardoor het trainen met verschillende parameters logischerwijs ook meer tijd in beslag zal nemen. Hier is helaas niet persé een mitigatie voor.

[4] https://cityflow-project.github.io/
[5] https://www.eclipse.org/sumo/
[6] https://github.com/projectmesa/mesa

Validatie van model

In het opzetten van de eisen van het programma wordt al besproken hoe de oplossing geëvalueerd moet worden. Men zou de huidige oplossing die momenteel gebruikt wordt voor het sturen van verkeerslichten kunnen gebruiken als een base-line. Er wordt dan gekeken hoe de huidige oplossing scoort op alle relevante metrics (zoals bijv. verkeersdoorstroom). Vervolgens na het trainen van een nieuwe oplossing zal er gekeken worden hoe deze scoort ten opzichte van de baseline. Scoort deze lager dan de baseline dan is de nieuwe oplossing dus nog niet goed genoeg.

Voor het laatste onderdeel wordt ingeschat 1 week nodig te hebben.

Conclusie

Op basis van literatuuronderzoek en experimentatie tijdens de laatste paar weken is er informatie opgedaan over de werking en gebruik van RL methodes. Er is destijds gevraagd om naar mogelijkheden om clusters verkeerslichten in de stad aan te sturen met RL te kijken. Dat de traffic-flow in de praktijk geoptimaliseerd kan worden met geen RL-gebaseerde oplossingen valt te concluderen uit bijvoorbeeld het SURTRAC systeem dat in Pittsburgh geïmplementeerd is en goede resultaten geeft.
Desalniettemin blijkt uit de literatuuronderzoek dat er in het verleden al goede resultaten behaald zijn in het verbeteren van de doorstroming met behulp van RL technieken (multi-agent Q-learning). Ook kwam hier naar voren dat clusters van verkeerslichten met elkaar kunnen communiceren om zo ook de doorstroming te bevorderen. Wat betreft veiligheid kunnen de acties zo opgezet worden dat de agent alleen maar de keuze heeft tussen veilige acties en kunnen er ook restricties ingebouwd worden die voetgangers en fietsers in bescherming nemen.
Op basis van de hierboven beschreven punten en de rest van dit adviesrapport kunnen wij dan ook concluderen dat wij wel degelijk mogelijkheden zien om clusters verkeerslichten in de stad aan te sturen met reinforcement learning en dat deze mogelijkheden haalbaar zijn.

Bronnen

Bibliography

Liu, Y., Liu, L., & Cheng, W. (2017). Intelligent Traffic Light Control Using Distributed Multi-agent Q Learning. https://arxiv.org/pdf/1711.10941.pdf

Nieuw realtime model voor verkeerslichten reduceert wachttijd en CO2 uitstoot (Sweco, Compiler). (z.d.). Sweco. https://www.sweco.nl/actueel/nieuws/nieuw-realtime-model-voor-verkeerslichten-reduceert-wachttijd-en-co2-uitstoot/

Noort, W. (2015, november 18). *Google en TNO gaan app-data gebruiken om files te voorkomen. NRC*. NRC. https://www.nrc.nl/nieuws/2015/11/19/google-en-tno-gaan-app-data-gebruiken-om-files-te-1558453-a205308

Schneider, C. (2020). *Deep Q-Learning in Traffic Signal Control: A Literature Review*. Retrieved 06 01, 2021, from https://repository.tudelft.nl/islandora/object/uuid:259debb3-4583-4bb1-9cd9-a8d5b88186e4/datastream/OBJ1/download

Smith, S. F., Barlow, G. J., Xie, X.-F., & Rubinstein, Z. B. (2013). Smart Urban Signal Networks: Initial Application of the SURTRAC Adaptive Traffic Signal Control System. In *Proceedings 23rd International Conference on Automated Planning and Scheduling (ICAPS '13)* (pp. 434 - 442). https://nacto.org/docs/usdg/smart_urban_signal_networks_smith.pdf

Speck, J. (2018). Understand Induced Demand. In *Walkable City Rules* (pp. 64-65). Island Press, Washington, DC. https://doi.org/10.5822/978-1-61091-899-2_27

Steinbuch, M. (2021, januari 3). *Smart Traffic Lights. Innovation Origins*. innovationorigins.com. https://innovationorigins.com/en/smart-traffic-lights/

The TF-Agents Authors. (2021, 01 28). *Introduction to RL and Deep Q Networks*. TensorFlow. Retrieved 05 31, 2021, from https://www.tensorflow.org/agents/tutorials/0_intro_rl

Wei, H., Zheng, G., Gayah, V., & Li, Z. (2020, December). Recent Advances in Reinforcement Learning for TrafficSignal Control: A Survey of Models and Evaluation. *ACM SIGKDD Explorations Newsletter*, *volume 22*, 7. https://doi.org/10.1145/3447556.3447565

Wei, H., Zheng, G., Yao, H., & Li, Z. (2018). IntelliLight. Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. *IntelliLight*. https://doi.org/10.1145/3219819.3220096

Wikipedia. (2021, Mei 12). *Traffic light control and coordination*. Wikipedia. Retrieved Mei 31, 2021, from https://en.wikipedia.org/wiki/Traffic_light_control_and_coordination