



Nationaal verkeerskundecongres 2016

BACK TO BASICS MET VERKEERSLICHTENREGELINGEN

Ruud van den Dries
MAP Traffic Management

Martijn Harmenzon
MAP Traffic Management

Jaap Vreeswijk
MAP Traffic Management

Samenvatting

Nederlandse verkeerslichtenregelingen zijn door de jaren heen te ingewikkeld, onoverzichtelijk en inflexibel geworden. Door een veelvoud van steeds nieuw toegevoegde functies en opties, die bovendien vaak zijn vervlochten door onderlinge afhankelijkheden, is het programmeren van regelingen dusdanig ingewikkeld geworden dat zeer uitgebreide specificaties nodig zijn en is veel specifieke domeinkennis nodig. Middels dit discussiepaper pleiten wij voor 'back to basics': een begrijpelijke en transparante verkeerslichtenregeling die geruime flexibiliteit biedt en met name geschikt is om technologische ontwikkelingen, nieuwe overtuigingen en nieuwe functies te accommoderen. Naar ons idee is de regeling open source beschikbaar, in een neutrale programmeertaal en modulair opgebouwd. Ons doel is om een discussie over een nieuwe generatie verkeerslichtenregelingen te initiëren. Wij zijn er van overtuigd dat dit op den duur zal leiden tot eenvoud en lagere kosten bij betere of gelijke effectiviteit, meer inzicht in de werking en prestatie van regelingen, nieuwe wijze van specificeren door wegbeheerders en bovenal een meer open markt.

Trefwoorden

Verkeerslichtenregeling, regelmethoden, Python, open

Inleiding, probleem en stelling

Nederlandse verkeerslichtenregelingen zijn door de jaren heen te ingewikkeld, onoverzichtelijk en inflexibel geworden. Door een veelvoud van steeds nieuw toegevoegde functies en opties, die bovendien vaak zijn vervlochten door onderlinge afhankelijkheden, is het programmeren van regelingen dusdanig ingewikkeld geworden dat zeer uitgebreide specificaties nodig zijn en is veel specifieke domeinkennis nodig.

Er worden hoge functionele eisen gesteld aan verkeerslichtenregelingen. Ze dienen niet alleen voertuigafhankelijk te regelen, maar bijvoorbeeld ook detectiefouten te herstellen, wachttijden te voorspellen, doelgroepen met prioriteit af te wikkelen, etc. Daarnaast zijn er tal van randvoorwaarden door beleidskeuzes, regels en overtuigingen (bv. met betrekking tot fasevolgorde, minimale/maximale tijden, ontruimingstijden, etc.) die een grote invloed hebben op het fundament van een regeling en daarmee zelfs doorslaggevend kunnen zijn voor de flexibiliteit en effectiviteit van de regeling. Bovendien hebben veel wegbeheerders vaak eigen eisen en wensen met betrekking tot een regeling. Een continue groeiend en veranderend eisenpakket heeft zondermeer bijgedragen aan de complexiteit van regelingen en geleid tot interne tegenstrijdigheden, maar vraagt tegelijkertijd ook om meer flexibiliteit (een tegenstrijdigheid op zichzelf).

Verkeerslichtenregelingen worden in Nederland, over het algemeen, in CCOL geprogrammeerd zijnde de standaard voor verkeerslichtenregelingen. Daarnaast zijn er een aantal (netwerk)regelingen op de markt die weliswaar zijn gebaseerd op nieuwe regelfilosofieën, maar welke voor buitenstaanders (zijnde anders dan de leverancier) nauwelijks zijn te doorgronden omdat de software en algoritmie vaak niet inzichtelijk is en niet open source is.

Middels dit discussiepaper pleiten wij voor 'back to basics': een begrijpelijke en transparante verkeerslichtenregeling die geruime flexibiliteit biedt en met name geschikt is om technologische ontwikkelingen, nieuwe overtuigingen en nieuwe functies te accommoderen. Naar ons idee is de regeling open source beschikbaar, in een neutrale programmeertaal en modulair opgebouwd. Wij stellen ons een verkeerslichtenregeling voor die bestaat uit drie niveaus:

1. Een basis- en noodregeling: een star regelprogramma waarop altijd teruggevallen kan worden;
2. Een basis voertuigafhankelijke regeling: dit is een slimme voertuigafhankelijke verkeersregeling waarin basis functionaliteiten zijn geborgd; en
3. Een bovenliggende schil: waardoor de basis regeling kan omgaan met diverse intelligente modules voor regeloptimalisaties.

In dit paper schetsen wij de bredere context en bespreken een aantal stappen die wij zelf al hebben gezet. Het is niet ons doel om een nieuwe concurrerende regeling op markt te brengen, maar om een discussie over een nieuwe generatie verkeerslichtenregelingen te initiëren. Wij zijn er van overtuigd dat dit op den duur zal leiden tot eenvoud en lagere kosten bij betere of gelijke effectiviteit, meer inzicht in de werking en prestatie van regelingen, nieuwe wijze van specificeren door wegbeheerders en bovenal een meer open markt.

Achtergrond

De tijd is rijp. Een aantal ontwikkelingen die de afgelopen jaren zijn ingezet geven blijk van een veranderende markt en werkwijzen. Bovendien bieden ze een basis voor de aanpak zoals in dit paper beschreven. Allereerst de Routekaart 2013-2023 'Beter geïnformeerd op weg' [1] waarin publieke en private partijen gezamenlijk een actieprogramma hebben geschetst wat onder meer zal leiden tot: een meer coöperatieve rol van wegwakantsystemen, een slimme mix van business-to-government, business-to-business en business-to-consumer, maximale openheid en beschikbaarheid van data en publiek-private samenwerking en allianties. In projecten, bijvoorbeeld A58 Spookfiles, PraktijkProef Amsterdam en The Call for Innovation Partnership Talking Traffic, heeft dit al concreet geleid tot nieuwe samenwerkingsvormen vaak op basis van afspraak gebaseerd op koppelvlakken en prestatie.

Een andere ontwikkeling, meer specifiek voor verkeerslichten, is de architectuur voor Intelligente Kruispunten (iVRI) [2]. Hierin worden componenten en koppelvlakken van verkeersregelautomaten (VRA) en coöperatieve wegkantstations (RIS) op zodanige wijze onderscheiden en gespecificeerd dat geen direct verband meer bestaat tussen de functionele eigenschappen van de componenten en waar deze fysiek zijn ondergebracht. Concreet houdt dit in dat ITS applicaties, bijvoorbeeld een regelapplicatie of een C-ITS applicatie, die traditioneel verbonden waren aan respectievelijk een VRA of RIS nu ergens anders kunnen worden ondergebracht en/of door een derde partij kunnen worden geleverd. Het zijn vooral de koppelvlakken tussen de zogeheten TLC Facilities en de ITS Applicaties (de TLC Facilities Interface) en de RSU Facilities en de ITS Applicaties (de RSU Facilities Interface) die deze transitie mogelijk maken. Mede door deze architectuur komen verkeersregeltechniek en C-ITS applicaties op een natuurlijkere wijze samen en kunnen deze integraal worden benut.

Een derde dimensie is de opkomst van infrastructuur-naar-voertuig communicatie en de intrede van zelfrijdende voertuigen. Doordat wegkantsystemen de verwachte schakelmomenten van verkeerslichten gaan communiceren en automobilisten/voertuigsystemen hier op zullen anticiperen moeten deze schakelmomenten voldoende voorspelbaar zijn. Hoewel er geen consensus is over de precieze eisen zullen regelingen waarschijnlijk voorspelbaarder moeten worden dan een typische voertuigafhankelijke regeling. Dat wil overigens niet zeggen dat een regeling niet meer flexibel en adaptief kan zijn. Anderzijds biedt voertuig-naar-infrastructuur communicatie ook nieuwe mogelijkheden voor verkeersregelalgoritmen, bijvoorbeeld door de additionele informatie die beschikbaar komt over naderende voertuigen. 'Floating Car Data' biedt continue detectie van voertuigen binnen het radiobereik van de RIS en levert naast de positie van voertuigen ook informatie als voertuigclassificatie, snelheid en mogelijk de geplande beweging op het kruispunt. En is nog weinig onderzoek gedaan naar nieuwe methoden die dergelijke informatie ten volle benut.

'Het nieuwe regelen', C-ITS ontwikkelingen en de mogelijkheden die ontstaan door een benadering als de iVRI architectuur zijn niet ongemerkt gebleven. Zo lanceerde 's-Hertogenbosch in samenwerking met Vialis begin 2016 als eerste een verkeersregeling die vanuit de Cloud wordt aangestuurd. Provincie Noord-Holland werkt samen met Nissan om de (on)mogelijkheden van directe communicatie van volledig voertuigafhankelijke verkeerslichten met zelfrijdende voertuigen te onderzoeken. Verder zijn er tal van universiteiten en onderzoeksinstituten die experimenteren met regelalgoritmes die feitelijk veronderstellen dat er een directe verbinding is met alle verkeer en er veel vrijheidsgraden zijn om tot een zeker optimum te komen. Massachusetts Institute of Technology (MIT) gaat zelfs zo ver dat ze een kruispuntregeling ontwikkelen die geen gebruik meer maakt van lantaarn maar enkel de trajectoriën van verkeer regelt [3].

Een anderen aanpak

Zoals aangegeven zijn de huidige regelingen vaak log en complex als het gaat om de code die is geschreven om de regeling goed te laten functioneren. Dit werpt veelal een barrière op waardoor opdrachtgevers en wegbeheerders zelf niet meer in staat zijn om de software voldoende te begrijpen en sterk afhankelijk blijven van de software leverancier of specialisten. Het kost veel tijd om de domeinkennis die nodig is op te doen alvorens iemand echt in staat is een regeling te programmeren die van dezelfde kwaliteit is als wanneer deze door een expert geprogrammeerd zou zijn. Om dit te doorbreken stellen wij een concept voor op basis van een begrijpelijke en een breed gedragen programmeertaal.

In ons concept hebben wij gekozen voor Python. Dit is een relatief jonge taal, die wereldwijd voor zeer veel uiteenlopende projecten gebruikt wordt en door veel programmeurs wordt beheerst. Veel meer dan het voor verkeerslichtenregelingen gebruikte C/COL. Hierdoor kan iedere programmeur, relatief snel, een verkeerslichtenregeling programmeren. Daarnaast is het voor een ieder mogelijk om nieuwe functionaliteit voor een regeling te programmeren en toe te voegen. Zo is er minder domeinkennis benodigd en is het voor een veel grotere groep belanghebbende mogelijk om de programmatuur te begrijpen en naar eigen inzicht en behoefte verder uit te breiden of aan te passen.

Middels het uit te werken concept willen wij een aantal zaken onderzoeken. Ten eerste willen wij onderzoeken of in Python een begrijpelijke, transparante regeling is te programmeren die voldoet aan de CROW richtlijnen. De eerste versie van de Python-regeling zal qua functionaliteit gelijk zijn aan de huidige regeling geprogrammeerd in CCOL. Daarbij is het uiteraard ook van belang om te bepalen of de kwaliteit van de regeling en het afwikkelingsniveau gelijkwaardig is. Vervolgens willen wij de werking van het concept van een modulaire regeling aantonen. De regeling beschikt dan over een aantal modules die elk een eigen functionaliteit aan de regeling toevoegen. Deze modules moeten naadloos op de regeling aansluiten, en onafhankelijk van elkaar en zonder verdere beperkingen voor de basisregeling kunnen werken. Ten slotte willen wij met de Python regeling, nieuwe regelprincipes uitproberen en evalueren. Denk hierbij aan 'n dynamische fasevolgorde, variabele geel-, groen- en ontruimingstijden en C-ITS ontwikkelingen zoals inputdata direct vanuit voertuigen (bv. snelheid en acceleratie) en ook terug levering van data direct aan voertuigen (bv. snelheid- en rijstrookadvies).

Naast dat de code zelf eenvoudiger en leesbaarder wordt willen wij een eenvoudig dashboard bieden. Daarmee moet het mogelijk worden om zonder programmeerkennis een regeling bij te sturen. Via het dashboard heeft de gebruiker inzicht in de prestatie van de regeling, eventuele interne tegenstrijdigheden als gevolg van de huidige instellingen (bv. conflicterende prioriteiten of maximale wachttijden) en de mogelijkheid om zelf aanpassingen te doen door gewenste waarden van Key-Performance-Indicators (KPIs) in te geven. De regelen bezit vervolgens zelf voldoende zelf-configurerende en corrigerende eigenschappen om hier onder water naar te acteren en de gebruiker via het dashboard terugkoppeling te geven.

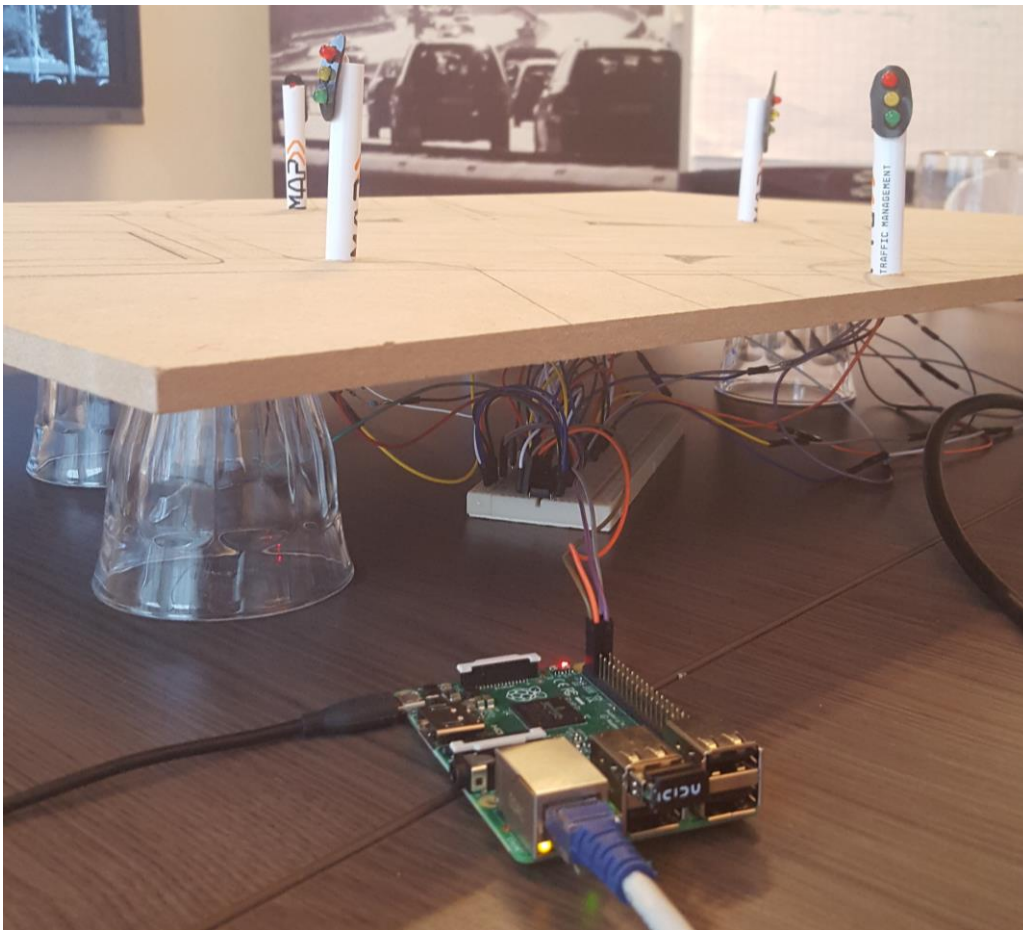
Samengevat: door een nieuwe verkeerslichtenregeling te ontwikkelen willen wij vooral aantonen dat het anders kan, dit tot nieuwe mogelijkheden leidt en kwaliteit verhogend is.

Om dit mogelijk te maken zijn wij begonnen met het programmeren van een regeling in Python. Aangezien wij de werking ook graag willen laten zien en behoefte hebben aan een proefomgeving om zaken in te testen, hebben we een schaalmodel gebouwd van een bestaande, met CCOL geregeld, kruispunt. De software draait op een RaspberryPi en stuurt alle ledjes individueel aan. Detectievelden en drukknoppen zijn beschikbaar in de vorm van schakelaars. Op termijn willen we ook data uitwisseling middels bijvoorbeeld wifi-p (wifi variant voor voertuig-wegkant communicatie) mogelijk maken. Voor meer informatie en illustraties zie [4,5].

In eerste instantie richten wij ons op een regeling voor één kruispunt. Hiermee willen wij aantonen dat een verkeerslichtenregelingen in Python en op basis van eerdergenoemde principes werkt. Daarna willen wij kijken naar de werking op netwerkniveau. Effectevaluaties van de regeling zullen worden gedaan met behulp van het open source simulatiepakket SUMO [6]. Daarmee kunnen wij bijvoorbeeld een vergelijkingstest doen, of complexere kruispunten en netwerken simuleren. Dit pakket werkt goed samen met Python, kent wereldwijd veel gebruikers en heeft diverse uitbreidingsmogelijkheden om bijvoorbeeld V2X communicatie te simuleren. Met de effectevaluaties willen we inzicht geven in de effecten van randvoorwaarden, eisen en keuzes op de complexiteit, interne tegenstrijdigheden en prestaties van een regeling.

Tot slot willen wij nogmaals benadrukken: het is niet ons doel om een nieuwe concurrerende regeling op markt te brengen, maar om een discussie over een nieuwe generatie verkeerslichtenregelingen te initiëren. Wij zijn er van overtuigd dat dit op den duur zal leiden tot eenvoud en lagere kosten bij betere of gelijke effectiviteit, meer inzicht in de werking en prestatie van regelingen, nieuwe wijze van specificeren door wegbeheerders en bovenal een meer open markt.

Graag maken wij van deze gelegenheid gebruik een ieder die interesse heeft om mee te denken en nieuwe ideeën uit te proberen met behulp van de gepresenteerde werkwijze!



Figuur 1: schaalmodel kruispunt met RaspberryPi

Referenties

1. Routekaart 2013-2023 'Beter geïnformeerd op weg'.
URL: <http://www.connectingmobility.nl/Over+ons/Routekaart/default.aspx>
2. iVRI, Deliverable F; iTLC Architecture, v1.2.
URL: <https://www.ivera.nl/downloads#beter-Benutten-ivri>
3. MIT Senseable City Lab.
URL: <http://senseable.mit.edu/wave/>
4. Een Nieuwe VRI; Hybride oplossing voor de toekomst
URL: <https://www.linkedin.com/hp/update/6159329687333072896>
URL: <http://dev.maptm.nl/mapvri/#visie>
5. "Simulation of Urban MObility" (SUMO)
URL: http://sumo.dlr.de/wiki/Main_Page