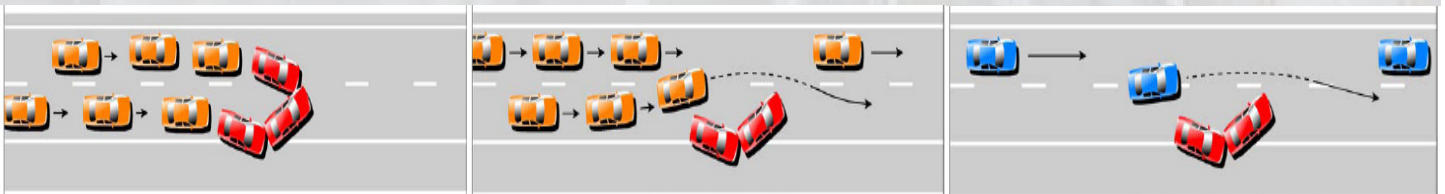


Automatische Incident Detectie met behulp van Meetlusgegevens

~ Onderzoek naar incidentdetectie op rijstrookniveau ~
~ Vergelijkingsstudie ~

mei 2007

Incident:
A random event that may disrupt the orderly flow of freeway traffic,
(Gall and Hall, 1989)





Automatische Incident Detectie met behulp van Meetlusgegevens

**~ Onderzoek naar incidentdetectie op rijstrookniveau ~
~ Vergelijkingsstudie ~**

Afstudeerscriptie
Leila R.W. van de Ven
Technische Universiteit Delft
Faculteit Civiele Techniek
Transport & Planning
Dynamisch verkeersmanagement

Commissie: Prof. dr. ir. S. Hoogendoorn (TU Delft)
Dr. Willem Jan Knibbe (AVV)
Dr. ir. H. van Lint (TU Delft)
Prof. dr. ir. B. De Schutter (TU Delft)
Ir. F. Middelham (AVV)
Ir. P. Wiggeraad (TU Delft)

Leila R.W. van de Ven
Boersestraat 3
5454 NG Sint Hubert
06-38305883
Leilavdven@hotmail.com
Mei 2007

Copyright © L.R.W. van de Ven 2007

Niets uit deze uitgave mag worden verveelvoudigd en/ of openbaar gemaakt worden door middel van druk, fotokopie, publicatie of op welke andere wijze dan ook zonder voorafgaande schriftelijke toestemming van de auteur. Voor het overnemen van gedeelten van deze publicatie voor studiedoeleinden verleent de auteur toestemming.

Colofon

Uitgegeven door:	ITS Edulab
Informatie:	L.R.W. van de Ven
Telefoon:	06 38305883 / 0049 2874 900379
Email	Leilavdven@hotmail.com
Uitgevoerd door:	L.R.W. van de Ven
Opmaak:	L.R.W. van de Ven
Datum:	mei 2007
Status:	definitief
Versienummer:	1

Inhoudsopgave

Voorwoord.....	6
Samenvatting	7
1. Inleiding.....	10
1.1 Wat is een incident.....	10
1.2 Incident Management.....	11
1.3 Waarom is een snelle detectie van incidenten van belang ...	11
1.4 Aanleiding / probleembeschrijving	12
1.5 Doel van het onderzoek.....	13
1.6 Opbouw van het onderzoek en het rapport	13
2. Verkeersdata en incidentdata.....	14
2.1 Verkeersdata.....	14
2.1.1. Het Rijkswaterstaat Monitoring Systeem	14
2.1.2. MoniGraph.....	15
2.1.3. Onderzoeksregio	16
2.1.4. Data incidenten	17
3. Verkeerskenmerken bij incidenten.....	18
3.1 Verkeerskundige gevolgen incident: schokgolftheorie.....	18
3.2 Verkeersafwikkeling als vraag groter dan restcapaciteit7	19
3.2.1. Vrije afwikkeling van het verkeer.....	20
3.2.2. Congestie	21
3.3 Verkeersafwikkeling als vraag kleiner dan restcapaciteit 7 ...	22
3.4 Welke verkeerskenmerken zijn te verwachten?	23
4. Beschrijving Incident detectie algoritmen.....	26
4.1 Incident detectie in de literatuur.....	26
4.2 Indeling Incident Detectie algoritmes	27
4.3 De Blokkadedetector	28
4.3.1. Het Algoritme.....	29
4.4 Presikhaaf algoritme.....	33
4.4.1. De werking van het algoritme.....	33
4.5 McMaster Algoritme	35
4.5.1. Bezettingsgraad.....	35
4.5.2. Het algoritme	36
5. Onderzoek bruikbaarheid rijstrookdata	41
5.1 De onderzochte incidenten	41
5.2 Analysemethode.....	41
5.3 Verdeling verkeer over rijstroken	42
5.4 Rijstrookverdelingen als referentie voor een incident	44
5.5 Onderscheid congestie en free flow.....	51

6. Incidentdetectie methode op rijstrookniveau.....	53
6.1 Uitgangspunten bij ontwikkelen detectiemethode op rijstrookniveau.....	53
6.2 Beschrijving detectie algoritme op strookniveau	53
7. Testen en vergelijken algoritmen.....	56
7.1 Indicatoren voor vergelijking	56
7.2 De incidenten	56
7.3 Implementatie van de detectiealgoritmen.....	58
7.3.1. Implementatie Blokkadedetector	58
7.3.2. Implementatie Presikhaaf algoritme	60
7.3.3. Implementatie McMaster algoritme	62
7.3.4. Implementatie Strook algoritme	63
7.4 Resultaten test	64
7.5 Vergelijkingen en conclusies	65
8. Conclusies en aanbevelingen.....	67
8.1 Conclusies	67
8.2 Aanbevelingen voor vervolgonderzoek	69
Referenties.....	71
Bijlagen	793

Voorwoord

Voor u ligt het eindrapport van het afstudeeronderzoek van L.R.W. van de Ven. Het onderzoek is uitgevoerd in het kader van het afronden van de Masterstudie Civiele Techniek aan de Technische Universiteit Delft, faculteit Civiele Techniek en Geowetenschappen, sectie "Transport en Planning". Binnen deze afstudeerrichting is de specialisatie gericht op dynamisch verkeersmanagement.

Het onderwerp van dit onderzoek, in opdracht van de Adviesdienst Verkeer en Vervoer (AVV) van Rijkswaterstaat, is vast te stellen of en hoe verkeersgegevens op rijstrookniveau bruikbaar zijn voor het detecteren van incidenten. Daarnaast is er de vraag of er andere algoritmes bruikbaar kunnen zijn in de Nederlandse situatie naast de huidige operationele Blokkadedetector. Het onderzoek is uitgevoerd binnen het ITS Edulab, een samenwerking tussen de Technische Universiteit Delft en de Adviesdienst Verkeer en Vervoer (AVV) van Rijkswaterstaat.

De auteur wil bij deze Prof. dr. ir. S. Hoogendoorn (TUDelft, Civiele Techniek) en Dr. Willem Jan Knibbe (AVV) bedanken voor hun intensieve begeleiding en deskundigheid tijdens mijn onderzoek. Verder wil ik ook de andere leden van de afstudeercommissie, Dr. Ir. H. van Lint (TUDelft, Civiele Techniek), Prof. dr. ir. B. De Schutter (TUDelft, DCSC), Ir. F. Middelham (AVV) en Ir. P. Wiggeraad (TUDelft, Civiele Techniek) bedanken voor hun bijdrage tijdens mijn afstudeertraject.

Delft, mei 2007

Leila R.W. van de Ven
Leilavdven@hotmail.com

Samenvatting

De Nederlandse wegen worden steeds meer overheerst door files. In veel gevallen gaat het om dagelijkse 'voorspelbare' congestie en is de oorzaak een te grote verkeersvraag voor een te lage capaciteit. Naast deze voorspelbare oorzaken op verwachte plaatsen en tijdstippen, kunnen files en vertragingen ook ontstaan door onvoorspelde oorzaken zoals pechgevallen, afgevallene ladingen of ongevallen. Deze worden ook wel aangeduid met de term incidenten en ontstaan op onverwachte plaatsen en tijdstippen. Juist omdat incidenten zo onvoorspelbaar zijn, is het van belang om deze in een zo vroeg mogelijk stadium te kunnen herkennen en maatregelen te nemen om de gevolgen zoveel mogelijk te beperken. De invoering van spitsstroken heeft de noodzaak tot een snelle detectie nog groter gemaakt. Tijdens de openstelling van een spitsstrook is er geen vluchtstrook beschikbaar. Het is dan van groot belang dat de spitsstrook zo snel mogelijk kan worden vrijgemaakt voor onder andere hulpdiensten. Door Rijkswaterstaat is Incident Management opgestart om de gevolgen van incident zoveel mogelijk te beperken en een goede doorstroming en veiligheid te waarborgen bij incidenten. Een snelle detectie van incidenten is de eerste stap in het verbeteren en versnellen van Incident Management en dus de gevolgen van incidenten te beperken.

Voor het automatisch detecteren van incidenten kan gebruik worden gemaakt van verschillende meetgegevens zoals video, meetlussen of rader. In dit onderzoek wordt alleen het detecteren van incidenten aan de hand van meetlusgegevens in beschouwing genomen. Deze meetlusgegevens bestaan uit intensiteiten en snelheden gemeten door lusdetectoren, die zich in paren in het wegdek bevinden op afstanden tussen 300 en 1000 meter. Elke minuut worden deze meetgegevens per meetluslocatie door het Rijkswaterstaat Monitoringsysteem beschikbaar gesteld in ADY (Afname Dynamische gegevens) minuut bestanden. Sinds enkele jaren zijn deze intensiteit- en snelheidsgegevens ook per rijstrook beschikbaar. De eerste vraag van dit onderzoek is dan ook *of het gebruik van data op rijstrookniveau toegevoegde waarde kan zijn voor incident detectie en hoe een dergelijk algoritme op rijstrookniveau eruit kan zien*. In de literatuur zijn vrijwel geen bronnen te vinden over incidentdetectie op rijstrookniveau.

Wel zijn er veel studies te vinden over verschillende ontwikkelingen en testen van detectiemethoden op rijbaanniveau. Een van de meest succesvol toegepaste methode in het buitenland is het McMaster algoritme (Gall en Hall, 1989; Hall et al., 1993). Het McMaster algoritme is ontwikkeld door de Traffic Research Group van de McMaster University van Toronto. Het algoritme zoekt naar congestievorming en controleert vervolgens of deze congestie is ontstaan door incidenten. In Nederland is de Blokkadedetector ontwikkeld en getest (Transpute, 2002). Dit algoritme zoekt naar typische intensiteits- kenmerken die zich voordoen tijdens een incident en voert vervolgens meerdere controles uit om met zekerheid vast te kunnen stellen dat het om een incident gaat. Naast deze methode is binnen Rijkswaterstaat het Presikhaaf algoritme ontwikkeld als alternatief op de Blokkadedetector. Dit algoritme zoekt naar typische

snelheidskenmerken die zich voordoen tijdens een incident. De Blokkadedetector is Nederland uitvoerig ontwikkeld en getest, maar functioneert niet optimaal. De resultaten van dit algoritme laten nog ruimte voor verbeteringen. De tweede vraag binnen dit onderzoek is dan ook *of er goede resultaten te behalen zijn uit de al bestaande McMaster Algoritme, het Presikhaaf algoritme en het nieuw te ontwikkelen algoritme op rijstrookniveau in vergelijking met de Blokkadedetector.*

Voor het onderzoek naar de mogelijkheden van incidentdetectie aan de hand van rijstrookgegevens zijn een aantal incidenten geanalyseerd. Hierbij is voornamelijk naar de verdeling van het verkeer over de rijstroken gekeken. In de verdeling van het verkeer over de rijstroken gedurende een dag is een locatieafhankelijk patroon zichtbaar afhankelijk van de rijbaanintensiteit. Tijdens incidenten is een duidelijke afwijking in de verdeling van het verkeer over de rijstroken waar te nemen die zicht in reguliere situaties niet voordoet. Dit geldt niet voor alle intensiteitsklassen. Bij lage rijbaanintensiteiten (ongeveer 545 vtg/uur per rijstrook) en zeer hoge rijbaanintensiteiten (ongeveer 1750 vtg/uur per rijstrook) kan niet worden gesproken van een afwijkende situatie tijdens incidenten. Ook kan uit de analyses van de incidenten worden opgemaakt dat de afwijkende verdeling van het verkeer, die te waarnemen is ten tijde van een incident, uitzonderlijk is en zich alleen voordoet ten tijde van het incident. Daarnaast is het rijstrookgebruik ten tijde van congestie anders dan in vrije afwikkeling.

Aan de hand van de bevindingen uit de incidentenanalyse is een eerste opzet gemaakt voor een detectiealgoritme op rijstrookniveau. Dit algoritme zoekt naar grote afwijkingen in rijstrookgebruik. Aan de hand van een aantal incidentvrije dagen worden voor elke locatie grenswaarden vastgesteld (aparte grenswaarden voor congestiesituaties en vrije afwikkeling) voor de verdeling van het verkeer over de rijstroken. Hiervoor worden per intensiteitsklasse alle minimale en maximale gemeten rijstrookaandelen als referentiewaarden genomen. Indien een gemeten rijstrookaandeel te veel afwijkt van de grenswaarden, wordt een melding van een incident gegeven. Zoals in de analyse van de incidenten kan worden opgemaakt geldt dit algoritme niet bij zeer lage of zeer hoge intensiteiten. Dit algoritme is samen met het Presikhaaf algoritme en het McMaster algoritme vergeleken met de Blokkadedetector.

Alle vier de algoritmes zijn in Matlab geïmplementeerd om ze te kunnen testen. Voor tests is gebruik gemaakt incidentmeldingen die door VCNL (Verkeerscentrale Nederland) zijn geregistreerd. De verkeersdata is afkomstig van de intensiteiten en snelheden uit de ADY minuutbestanden. De tests zijn voor een periode van twee maanden (juni en juli 2006) op de A12 Links in het district Zuid-Holland uitgevoerd.

Voor het vergelijken van de resultaten van de verschillende methodes is gelet op een drietal prestatie indicatoren:

- *Detectietijd*: de detectietijd is gelijk aan het verschil tussen de starttijd van het incident zoals deze staat aangegeven in de lijsten van VCNL en de tijd waarop het detectiealgoritme voor het eerst een melding geeft. Een negatieve detectietijd houdt dus in dat de methode het incident eerder meldt dan de melding bij de verkeerscentrales is binnengekomen.
- *Volledigheid*: de volledigheid is het percentage incidenten dat door

de methode juist is gemeld (dus zonder valsmeldingen) in verhouding tot het aantal incident dat VCNL in die periode meldt.

- *Aantal valsmeldingen:* het aantal valsmeldingen is het aantal keer dat het algoritme een melding van een incident geeft, terwijl er geen incidentmelding voorkomt in de lijsten van VCNL.

De prestaties van de verschillende methoden lopen erg uiteen. Op het McMaster algoritme na detecteren de methoden de incidenten gemiddeld eerder dan de gerapporteerde incidenten van VCNL. Dit houdt dus in dat automatische incident detectie een toegevoegde waarde heeft op het gebied van incidentdetectie.

Op basis van de resultaten van deze test kan worden geconcludeerd dat zowel het McMaster algoritme als het Strook algoritme (voor alle locaties) geen goede resultaten geven. Beide methodes detecteren een hoog percentage incidenten (het McMaster algoritme zelfs alle incidenten), maar het aantal valse meldingen is ook zeer hoog (beide methoden meer dan 14 per dag). Door deze hoge aantallen valsmeldingen zijn deze methodes niet acceptabel voor praktische toepassing. Bij het Strook algoritme daalt het aantal valsmeldingen aanzienlijk (van 703 naar 109) indien alleen naar de resultaten wordt gekeken van locaties met minder dan vier rijstroken. Een totaal van 109 valsmeldingen (2,2 per dag) is weliswaar nog niet acceptabel voor praktische toepassing, maar dit algoritme functioneert zeker niet optimaal. Het aanscherpen van de parameterinstellingen en het wegnemen van de vereenvoudigingen bij deze methodes zal betere resultaten opleveren. Een voordeel van het Strook algoritme boven de andere algoritmes is dat dit algoritme meer informatie over het incident geeft, bijvoorbeeld welke strook of stroken het betreft. Het Presikhaaf algoritme en de Blokkadedetector leveren beide goede resultaten als wordt gekeken naar de betrouwbaarheid. Beide algoritmes hebben een laag aantal valse meldingen. De blokkadedetector geeft gemiddeld eens in de twee dagen een valsmelding en het Presikhaaf algoritme over de gehele testperiode in totaal maar 1 valsmelding. Deze aantallen zijn acceptabel om praktische toe te passen.

Kort kan worden gezegd dat alle methodes bewezen hebben, toepasbaar te zijn voor incident detectie. Echter detecteert het Presikhaaf algoritme een te laag percentage incidenten en zal dit algoritme alleen toegevoegde kunnen hebben door de snelle detectie als deze in combinatie met andere detectiealgoritmes werkzaam is. De Blokkadedetector geeft redelijke resultaten, maar de resultaten zijn niet optimaal en laten nog veel ruimte voor verbeteringen. Het McMaster algoritme geeft in de tests geen goede resultaten, maar door het wegnemen van vereenvoudigingen en beter instellingen van parameters zullen de resultaten van dit algoritme wellicht verbeterd worden. Aangezien dit algoritme alle incidenten detecteert wordt nader onderzoek naar dit algoritme dan ook aanbevolen. Ook voor het Strookalgoritme wordt nader onderzoek aanbevolen. De resultaten van de tests zijn niet optimaal, voornamelijk op locaties waar de rijbaan uit meer dan drie rijstroken bestaat. Het Strook algoritme zal zeker beter kunnen gaan presteren als vereenvoudigen en onzekerheden weggenomen worden. Nader onderzoek naar dit algoritme zal voornamelijk liggen in betere instellingen van parameters en het verlagen van de gevoeligheid met betrekking tot meetfouten en afwijkende locaties.

1. Inleiding

Het eerste hoofdstuk van het onderzoeksrapport 'incident detectie' geeft een inleiding in het onderzoek en de opzet van het onderzoek weer. In de eerste paragraaf wordt ingegaan op de vraag wat incident detectie inhoudt en waarom het detecteren van incidenten van belang is. In paragraaf 1.2 wordt een probleembeschrijving gegeven. Vervolgens wordt in paragraaf 1.3 ingegaan op de probleemstelling en doelstelling. In de laatste paragraaf wordt de structuur van het onderzoek rapport toegelicht.

1.1 Wat is een incident

De Nederlandse wegen worden steeds meer overheerst door files. Alleen al in 2006 steeg de filedruk met 7,2 procent ten opzichte van 2005 (bron: VID). Veel van de oorzaken van deze files zijn te vinden in een te grote vraag en te lage capaciteit, voornamelijk bij op- en afritten of andere knelpunten. Op deze plaatsen ontstaat vaak de dagelijkse 'voorspelbare' congestie. Andere gevallen waneer er *voorspelbare* congestie zal ontstaan, is tijdens wegwerkzaamheden of evenementen. Deze oorzaken zorgen voor het belemmeren van een vlotte doorstroming van het verkeer op **verwachte** plaatsen en tijdstippen.

Figuur 1.1:

Incidenten blokkeren vaak lang de weg en leiden tot lange files op onverwachte plaatsen en tijdstippen



Naast voorspelbare oorzaken van files zijn er ook oorzaken die onvoorspelbaar zijn, zoals pechgevallen, ongevallen of afgevalen ladingen. Deze worden ook wel aangeduid met de term 'incidenten'. Gall and Hall (1989) definieerden een incident als: *a random event that may disrupt the orderly flow of freeway traffic*. Incidenten zijn dus verantwoordelijk voor het belemmeren van een vlotte doorstroming van het verkeer en ontstaan op onverwachte plaatsen en tijdstippen. Juist omdat incidenten zo onvoorspelbaar zijn is het van belang om deze in een zo vroeg mogelijk stadium te kunnen herkennen en maatregelen te nemen om de gevolgen zoveel mogelijk te beperken.

1.2 Incident Management

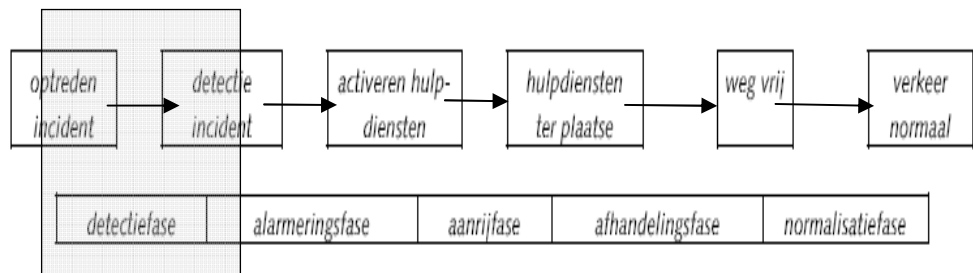
Al jaren wordt er veel aandacht besteed aan het bestrijden van files. Voor de verwachte en dagelijkse files zijn er onder andere maatregelen als toeritdosering en informatie aan weggebruikers over de verkeersomstandigheden. Voor het bestrijden van onverwachte files die ontstaan door incidenten heeft Rijkswaterstaat samen met hulpdiensten zoals de politie en brandweer Incident Management opgestart. Incident Management heeft als doel files op het Rijkswegennet, die ontstaan door incidenten drastisch te verkorten om zowel de doorstroming en veiligheid te waarborgen. Met behulp van een aantal technische maatregelen, zoals het gebruik van verkeerssignalering en camerabewaking wordt de doorstroming van het verkeer bevorderd. Door het werken aan een optimale samenwerking tussen hulpdiensten, zoals een verbeterde uitvraagprocedure bij een melding en het instellen van een coördinatieteam van vertegenwoordigers van alle hulpdiensten bij een incident wordt het afhandelingsproces bij incidenten geoptimaliseerd.

Incident Management is op te splitsen in vijf fasen: detectiefase, alarmeringsfase, aanrijfase, afhandelingsfase en normalisatiefase. In onderstaande figuur staan deze fase weergegeven.

.....

Figuur 1.2:

De vijf fasen van Incident Management



Dit onderzoek richt zich op een specifiek pad in de detectiefase, dit is de periode tussen het tijdstip van optreden van een incident en het tijdstip dat het incident wordt geregistreerd in een centrale. Het gaat hierbij om het automatisch detecteren met behulp van verkeersgegevens uit detectielussen. Een snelle detectie van incidenten is de eerste stap in het verbeteren en versnellen van Incident Management.

1.3 Waarom is een snelle detectie van incidenten van belang

Hoe sneller incident management kan verlopen, hoe minder erg de gevolgen van incidenten zullen zijn: minder vertragingen, verhoging van de veiligheid, minder kosten, minder tijdsbeslag voor politie, hulpverleners en ambulances door betere informatie. Zoals hierboven beschreven is de detectie van incidenten de eerste stap binnen Incident Management die ervoor kan zorgen dat het gehele proces verbeterd en versneld wordt. Door een snelle detectie worden operators in staat gesteld snel maatregelen te nemen zoals het afkruisen van stroken en het inschakelen van hulpdiensten en levert een snelle detectie zowel een bijdrage aan het beperken van congestie als aan de veiligheid.

De invoering van spitsstroken heeft de noodzaak tot een snelle detectie van incidenten nog groter gemaakt. Een spitsstrook is een vluchtstrook die tijdens de spitsuren ingezet wordt als tijdelijke extra rijstrook en buiten de spitsuren

een normale functie van een vluchtstrook vervult. Gedurende de openstelling van de spitsstroken is er dus geen vluchtstrook beschikbaar. Vanuit veiligheidsoverwegingen is het van belang dat de vluchtstrook bij een incident weer snel vrij gemaakt kan worden voor onder andere hulpdiensten. Des te sneller een incident dus gedetecteerd kan worden, des te sneller de vluchtstrook vrij gemaakt kan worden.

Figuur 1.3:

Steeds meer vluchtstroken worden op drukke tijden ingezet als spitsstroken, wat de noodzaak tot een snelle detectie van incidenten vergroot.



1.4 Aanleiding / probleembeschrijving

De aanleiding voor dit onderzoek komt voort uit de noodzaak tot een steeds snellere en betrouwbaardere detectie van incidenten, voornamelijk ook voor de spitsstroken. Zoals hierboven beschreven, wordt de vluchtstrook steeds vaker ingezet als rijstrook. Tijdens spitsen is het van groot belang dat de weg zo snel mogelijk kan worden vrijgemaakt en het incident toegankelijk is voor hulpdiensten. De noodzaak tot een snelle, maar ook betrouwbare melding van incidenten wordt dus steeds groter. De huidige detectiemethode die in Nederland in gebruik is, de blokkadedetector, laat nog ruimte voor verbetering. Een van deze verbetering zou kunnen liggen in het gebruik van data op rijstrookniveau in plaats van rijbaanniveau. Deze data is sinds enkele jaren toegankelijk en kan mogelijk voor verbeteringen zorgen in de detectiefase. In de literatuur zijn vrijwel geen bronnen te vinden over incidentdetectie op rijstrookniveau. Wel zijn er veel studies te vinden over verschillende ontwikkelingen en testen van detectiemethoden op rijbaanniveau (Martin et al. 2001, Rakha et al. 2004). Een van deze meest succesvol toegepaste methode in het buitenland is het McMaster algoritme (Gall en Hall, 1989; Hall et al., 1993). Tot dusver is nog geen onderzoek gedaan of dit algoritme ook in Nederland tot goede resultaten kan leiden. Ook is er door Rijkswaterstaat het Presikhaaf algoritme ontwikkeld als alternatief van de blokkadedetector. Het onderzoeken van een aantal bestaande algoritmes geeft dit onderzoek een toegevoegde waarde om de kennis, inzicht en mogelijkheden van incident detectie algoritmes te vergroten en tevens te onderzoeken of deze methoden toegevoegde waarde kunnen hebben op het detecteren van incidenten in Nederland.

1.5 Doel van het onderzoek

Het hoofddoel van dit onderzoek is tweeledig en kan als volgt geformuleerd worden:

1. Kan het gebruik van data op rijstrookniveau toegevoegde waarde hebben voor incident detectie en hoe kan een dergelijk algoritme op rijstrookniveau eruit zien
2. Zijn er goede resultaten te behalen uit de al bestaande McMaster Algoritme, het Presikhaaf algoritme (ontwikkelt binnen RWS) en het nieuw te ontwikkelen algoritme op rijstrookniveau in vergelijking met de Blokkadedetector.

1.6 Opbouw van het onderzoek en het rapport

In dit hoofdstuk is een inleiding in het onderzoek weergegeven en de opzet van het onderzoek. In hoofdstuk 2 wordt ingegaan op de data, zowel verkeersdata als incidentdata en middelen die gebruikt worden voor het onderzoek. In hoofdstuk 3 worden de theorie betreffende incidenten besproken. Hierin wordt aan de hand van schokgolftheorie inzicht gegeven in de te verwachten verkeerskenmerken bij incidenten op zowel rijbaan als rijstrookniveau. In hoofdstuk 4 wordt kort ingegaan op de belangrijkste zaken omtrent incident detectie in de literatuur. Daarnaast wordt in detail ingegaan op de werking van een drietal al bestaande algoritmes: de Blokkadedetector, het Presikhaaf algoritme en het McMaster algoritme. In hoofdstuk 5 wordt onderzocht of en hoe strookdata bruikbaar kan zijn voor het detecteren van incidenten. Hiervoor worden een aantal incidenten nader geanalyseerd aan de hand van beschikbare verkeersdata. Hoofdstuk 6 geeft een beschrijving van een mogelijke methode om incidenten te detecteren aan de hand van strookgegevens. In hoofdstuk 7 worden de besproken incidentmethoden en de nieuw ontwikkelde detectiemethode (op basis van strookgegevens) geïmplementeerd, op kleine schaal getest en geanalyseerd. In het laatste hoofdstuk worden de conclusies en aanbevelingen voor nader onderzoek gegeven.

2. Verkeersdata en incidentdata

Dit hoofdstuk geeft een beschrijving van de gebruikte gegevens, zowel de verkeersdata als de incidentgegevens. De verkeersdata betreft de minuutgegevens uit detectielussen in Monica (Monitoring Casco) afkomstig van het Rijkswaterstaat Monitoring Systeem. De gegevens over incidenten zijn afkomstig van het Verkeerscentrum Nederland van Rijkswaterstaat (VCNL)¹.

2.1 Verkeersdata

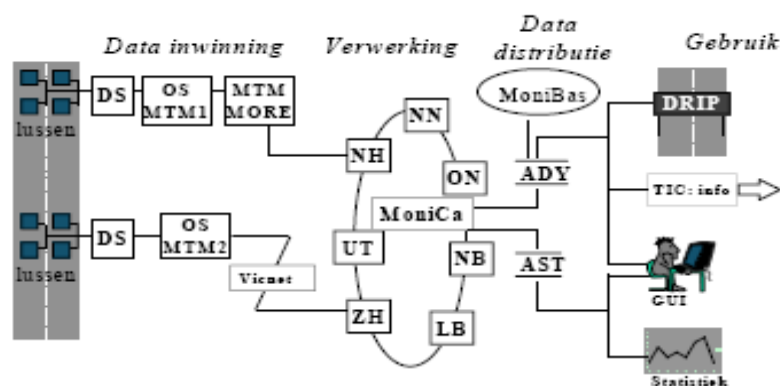
De gebruikte verkeersdata is afkomstig uit Monica. Monica is een onderdeel van het Rijkswaterstaat Monitoring Systeem.

2.1.1. Het Rijkswaterstaat Monitoring Systeem

Het Nederlandse (hoofd)wegennet is uitgerust met een uitgebreid monitoringsysteem, het Rijkswaterstaat Monitoring Systeem. Dit verkeersmanagement systeem is een ingewikkeld netwerk van data inwinning, data verwerking, distributie van data en apparatuur voor presentatie en gebruik. In figuur 2.1 is een vereenvoudigde weergave van het RWS Monitoring Systeem te zien.

Figuur 2.1:

Vereenvoudigde weergave van het Rijkswaterstaat Monitoring Systeem.
Bron:[8]



- Inwinnen van verkeersgegevens begint op de rijbaan, waar lusdetectoren in paren worden gebruikt die zich op afstanden tussen 300 en 1000 meter in het wegdek bevinden.
- Het detectorstation (DS) meet op basis van de lusgegevens de snelheid en intensiteit van de voertuigstromen op elke rijstrook of rijbaan. De detectorstations bevinden zich langs de kant van de weg in de nabijheid van de detectielusparen.
- Het DS geeft vervolgens elke minuut deze gegevens door aan alle op het DS aangesloten onderstations (OS) en informeert tevens over het

¹ VCNL is een onderdeel van Rijkswaterstaat en verzamelen alle informatie op landelijk niveau over de situatie op de Nederlandse wegen

-
- functioneren van de aangesloten detectielusparen.
 - Via het met MTM-2 (Motorway Traffic Management, versie 2) worden de gegevens doorgestuurd naar de zeven regionale MoniCa (MONItoring Casco) systemen. Het MTM-2 is een verkeerssignaleringsysteem ontwikkeld door RWS voor het bewaken van verkeersstromen en het waarschuwen van weggebruikers als er zich bijzondere situaties voordoen. Het is een volledig geautomatiseerd netwerk managementsysteem voor het hoofdwegennet en kent koppelingen met andere systemen zodat weggebruikers geen conflicterende instructies krijgen. Het MTM-2 wordt gebruikt voor applicaties als snelheidsmaatregelen, aansturing van onder meer toeritdosering en beveiliging van defecte infrastructuur situaties.
 - Vanuit het MTM-2 worden de verkeersgegevens en signaleringgegevens doorgestuurd naar MoniCa. Deze MoniCa systemen zijn met elkaar verbonden en wisselen hun informatie uit, zodat alle informatie in elke regio beschikbaar is. Na enkele technische controles stelt MoniCa de resultaten (dynamische en statische verkeersgegevens over intensiteiten, gemiddelde snelheden, voertuigcategorieën, files, spitsdrukte en spookrijdermeldingen) beschikbaar in de vorm van ADY (Afname Dynamische gegevens) minuut bestanden.
 - De gegevens die Monica levert, worden op hun beurt door andere applicaties gebruikt. Monibas (Monitoring Basisapplicaties) levert "opgewerkte" informatie zoals reistijd en filelengte. Deze informatie kan gebruikt worden voor de verkeersinformatie of om op DRIP's te tonen. De aansturing van de DRIP gaat met een apart stuursysteem (CDMS), dat Monica gegevens gebruikt. In de verkeerscentrale of online kunnen de Monica gegevens omgezet worden naar grafische plaatjes. Daarnaast worden de wegverkeerleiders in de verkeerscentrale geholpen met BOSS (Beslissing OnderSteunend Systeem). Astrid (Afname Statistische gegevens Richting Intens Database) zorgt voor het opslaan van statistische verkeersgegevens.

Voor dit onderzoek wordt gebruik gemaakt van de data die geleverd wordt door MoniCa. Deze data bestaat uit minuutgegevens over de intensiteit en snelheid voor elke locatie en voor elke rijstrook afzonderlijk.

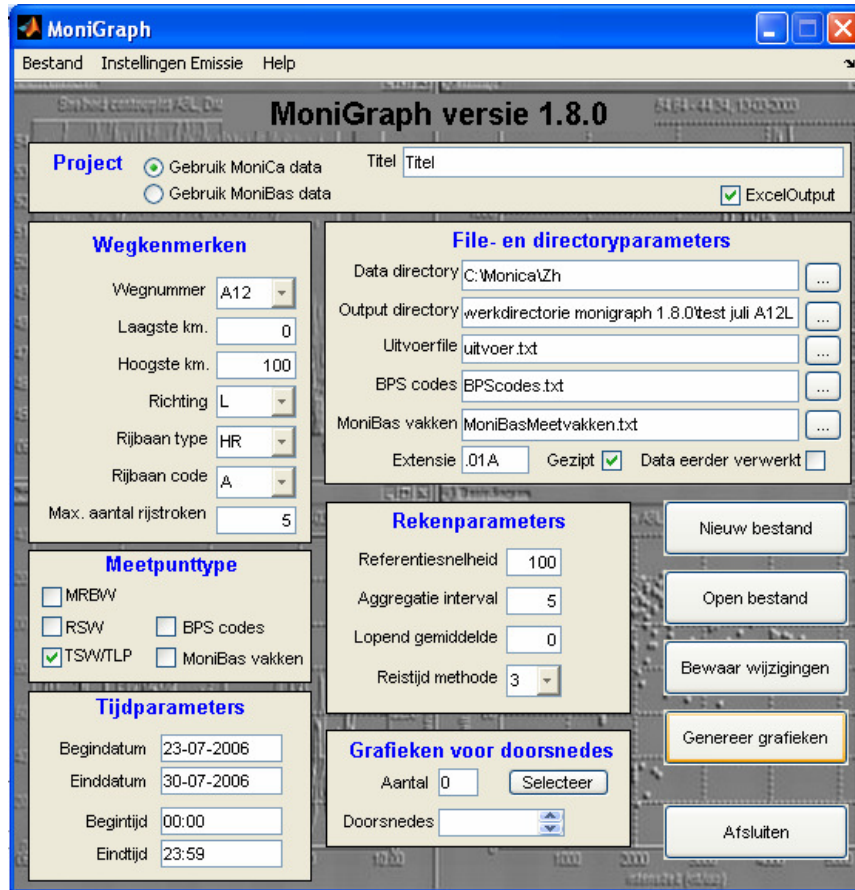
2.1.2. MoniGraph

De minuutfiles uit MoniCa worden aangeleverd in zogenoemde ADY-files. Deze files worden met een vertraging door de verwerking van ongeveer een minuut, elke minuut beschikbaar gesteld per regio. De ADY-files bestaan uit een code waarin onder ander het wegnummer, rijstrook en richting en de bijbehorende intensiteits- en snelheidsgegevens staan weergegeven. Voor het inlezen van deze files is het programma MoniGraph ontwikkeld door Henk Taale van AVV. Dit programma, gecodeerd in Matlab, verwerkt de gecodeerde data uit de ADY-files tot bruikbare en gevisualiseerde output.

Binnen het programma MoniGraph kan een invoerfile worden aangeroepen, zie figuur 2.2, waarin kan worden aangegeven welke data men beschikbaar wil hebben. Elke datum, tijdstip, wegnummer, richting en gewenste kilometersering kan worden ingegeven, waarna MoniGraph de benodigde codes uit de ADY-files haalt en deze verwerkt tot bruikbare visuele en tekstuele data. Hierbij geeft dit programma ook een melding indien er sprake is van ontbrekende en foutieve data.

Figuur 2.2:

Invoerfile voor MoniGraph, waarin kan worden aangegeven welke data je beschikbaar wilt hebben.



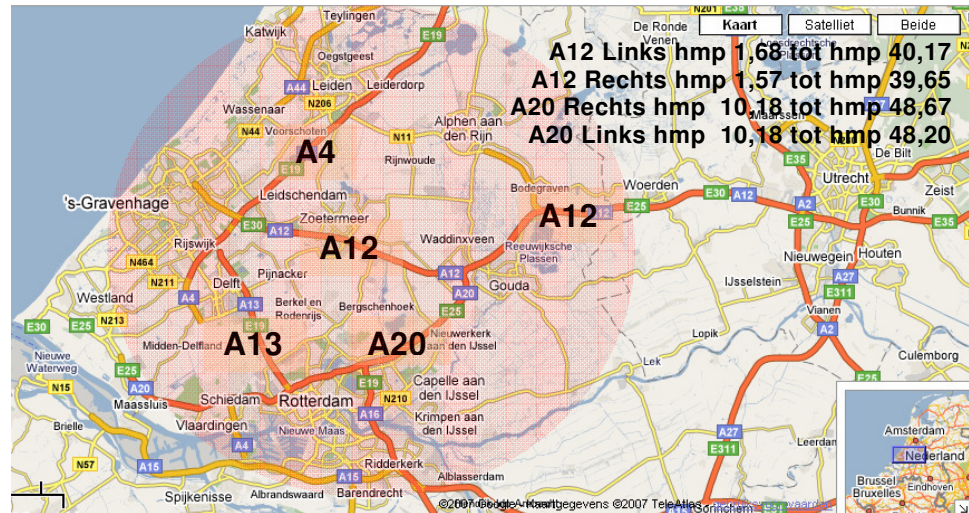
Binnen MoniGraph bestaat de mogelijkheid om zelf scripts toe te voegen, zoals incident detectie algoritmes. De benodigde data over intensiteiten en snelheden voor elke minuut en locatie en waar nodig ook voor elke rijstrook afzonderlijk, kan eenvoudig binnen MoniGraph worden aangeroepen.

2.1.3. Onderzoeksregio

Binnen MoniGraph kan per regio data verwerkt worden. Voor dit onderzoek is gebruik gemaakt van de regio Zuid-Holland voor de maanden juni en juli 2006. Tot het district Zuid-Holland behoren de autosnelwegen A4, A12, A13 en de A20 (zie onderstaande figuur).

Figuur 2.3:

De wegen dit tot het district Zuid-Holland behoren



2.1.4. Data incidenten

De gebruikte incident gegevens zijn afkomstig van het Verkeerscentrum Nederland (VCNL) van Rijkswaterstaat. VCNL bestaat sinds 2004 als het onderdeel van Rijkswaterstaat en is verantwoordelijk voor de landelijke taken op het gebied van operationeel verkeersmanagement. Ze verzamelen informatie over de actuele verkeerssituaties en zorgen dat via andere marktpartijen deze informatie bij de weggebruikers terecht komt. Het VCNL rapporteert en archiveert ook gegevens over de verkeerssituatie op de Nederlandse wegen. Indien er sprake is van geen normale doorstroming van het verkeer wordt hier een melding van gemaakt. Ook incidentmeldingen worden op deze wijze gerapporteerd. Onder andere de datum, locatie, begin en eindtijdstip, de oorzaak en filelengte van het incident wordt beschreven. Er wordt geen gedetailleerde omschrijving of informatie over de afsluiting van rijstroken tijdens het incident gegeven. De melding van de incidenten worden samengesteld uit meldingen die binnenkomen vanuit de meldcentrales. Hierdoor kunnen er onnauwkeurigheden ontstaan in voornamelijk de tijdstippen van de incidenten en de precieze locaties. Ook is het mogelijk dat bijvoorbeeld kleine incidenten niet worden gemeld aan de meldkamer en dus ook niet worden opgenomen in de incidentmeldingen van VCNL.

In dit onderzoek is gebruikt gemaakt van de incidentmeldingen die door VCNL zijn gerapporteerd voor de maanden juni en juli in het district Zuid-Holland voor de A12 en de A20. In bijlage A is een overzicht te zien van de incidentmeldingen in de maanden juni en juli voor de A12 en de A20.

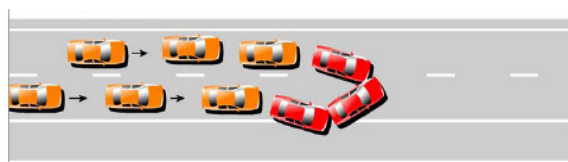
3. Verkeerskenmerken bij incidenten

Voordat ingegaan wordt op algoritmes om incidenten te detecteren is het eerst van belang om meer inzicht te krijgen in het gedrag van het verkeer tijdens incidenten. In dit hoofdstuk worden de theoretische verkeerskenmerken die bij incidenten op rijbaanniveau en rijstrookniveau worden verwacht besproken.

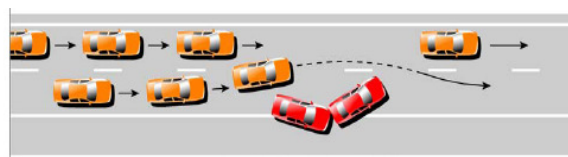
Gevolgen van incident zijn niet in alle situaties duidelijk herkenbaar uit verkeersdata. In onderstaand figuur is dit geïllustreerd. Als een incident zorgt voor een volledige blokkade van de weg of als de intensiteit na het incident hoger is dan de restcapaciteit, situatie A en B, zullen de verkeerskenmerken duidelijk herkenbaar zijn. Indien situatie C ontstaat, waar de intensiteit lager is dan de restcapaciteit of dat de blokkade van zeer korte duur is, zullen de verkeerskenmerken niet of nauwelijks herkenbaar zijn. Beide situaties worden apart behandeld in respectievelijk paragraaf 3.2 en 3.3.

Figuur 3.1:

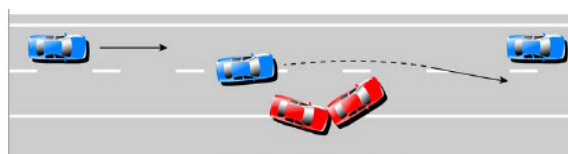
Een incident zorgt niet altijd voor een gehele blokkade van de weg en is afhankelijk van de capaciteit en intensiteit duidelijk of minder duidelijk herkenbaar. (bron: Transpute, 2004)



A. Weg geheel geblokkeerd:
Verkeerskenmerken duidelijk herkenbaar



B. Weg gedeeltelijk geblokkeerd, intensiteit hoger dan capaciteit
Verkeerskenmerken duidelijk herkenbaar



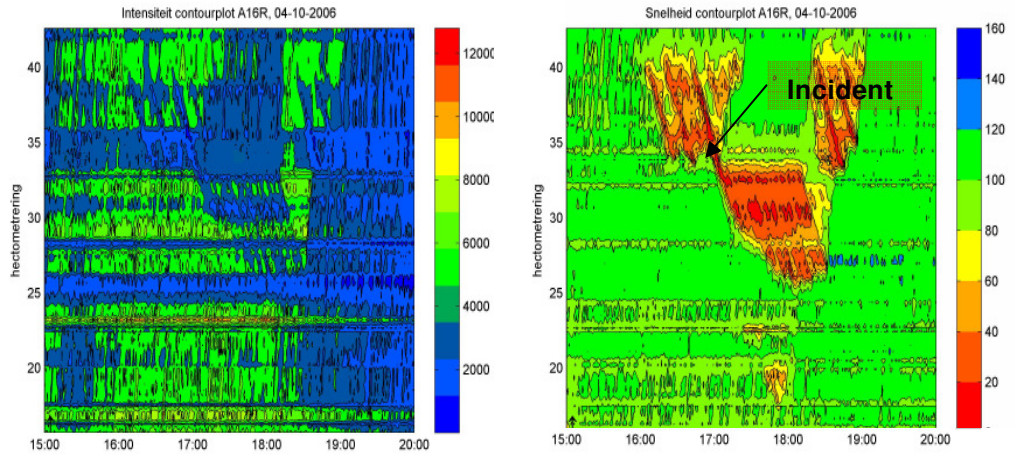
C. Weg gedeeltelijk geblokkeerd, intensiteit lager dan capaciteit:
Verkeerskenmerken niet of onduidelijk herkenbaar

3.1 Verkeerskundige gevolgen incident: schokgolftheorie

Een incident is een onvoorspelde oorzaak die voor filevorming kan zorgen. Ze ontstaan op onverwachte plaatsen en tijdstippen en zijn verantwoordelijk voor het belemmeren van een vlotte doorstroming van het verkeer. In onderstaande figuur is een intensiteitscontourplot en snelheidscontourplot uit MoniGraph van een incident weergegeven. Het betreft hier een ongeval op de A16. Rond 17:00 uur is de weg voor ongeveer 1,5 uur afgesloten geweest.

Figuur3.2:

Intensteits (vtg/uur) en snelheids (km/h) contourplot van een incident. Het betreft hier een ongeval waarbij de A16 rond 17:00 uur ongeveer 1,5 uur afgesloten is geweest. Er ontstond een file van 7 kilometer en het zorgde voor een vertraging van ongeveer 20 minuten.



In bovenstaande figuren is een duidelijk herkenbaar patroon zichtbaar in de snelheden en intensiteiten die zich voordoen een (deels) blokkerend incident. Of dit patroon ook daadwerkelijk herkenbaar zal zijn, is afhankelijk van de intensiteit van het verkeer en de capaciteit van de weg ten tijde van het incident. Met behulp van schokgolftheorie kan de verkeersafwikkeling in tijd en afstand goed worden weergegeven. Een schokgolf is een grens tussen twee verkeers toestanden die een positieve, negatieve of een snelheid van nul kan hebben. De snelheid van deze schokgolf kan worden berekend met:

$$\omega = \frac{q_2 - q_1}{k_2 - k_1} = \frac{\Delta q}{\Delta k} \quad [3.1]$$

waarin: ω = snelheid van de schokgolf [km/uur]

Δq = verschil in intensiteiten q_1 en q_2 tussen twee verkeers toestanden [vtg/uur]

Δk = verschil in dichtheden k_1 en k_2 tussen twee verkeers toestanden 1 en 2 [vtg/km]

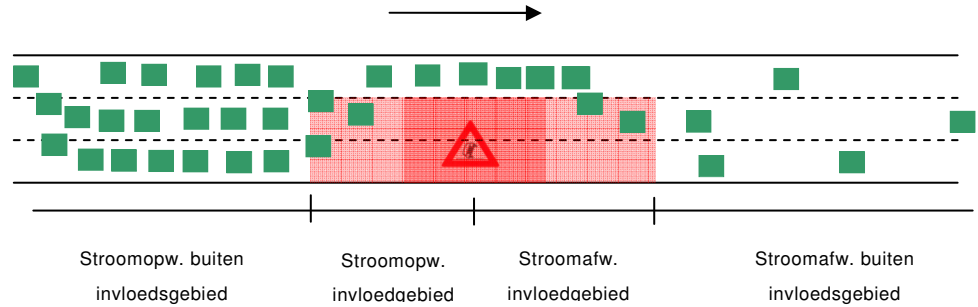
Voor meer informatie over de verkeersstroomtheorie en schokgolftheorie wordt verwezen naar bijlage B.

3.2 Verkeersafwikkeling als vraag groter dan restcapaciteit⁷

Wanneer een incident de capaciteit van de weg waarop het zich voordoet zodanig verkleint dat de gereduceerde capaciteit lager is dan de intensiteit (zie figuur 3.3), zal zich een file vormen. Er arriveren dan dus op een incidentlocatie meer voertuigen binnen een tijdseenheid dan er kunnen vertrekken. Zolang de vraag hoger ligt dan de capaciteit, zal de file blijven groeien. In dat geval is er geen vrije afwikkeling van het verkeer meer en zullen zowel stroomopwaarts als stroomafwaarts de effecten van de verlaagde capaciteit waarneembaar zijn.

Figuur 3.3:

Verkeersafwikkeling indien restcapaciteit na het incident **kleiner** is dan verkeersvraag



Voordat zich een incident voordoet kan er sprake zijn van een vrije afwikkeling of van congestie door bijvoorbeeld een toerit of wegversmalling. In het eerste geval zal een file ontstaan door het incident en in het tweede geval zal de file als gevolg van het incident toenemen. De verkeerscondities die verwacht worden zijn in beide gevallen verschillend. In subparagraaf 3.2.1 en respectievelijk 3.2.2 wordt hier verder op ingegaan.

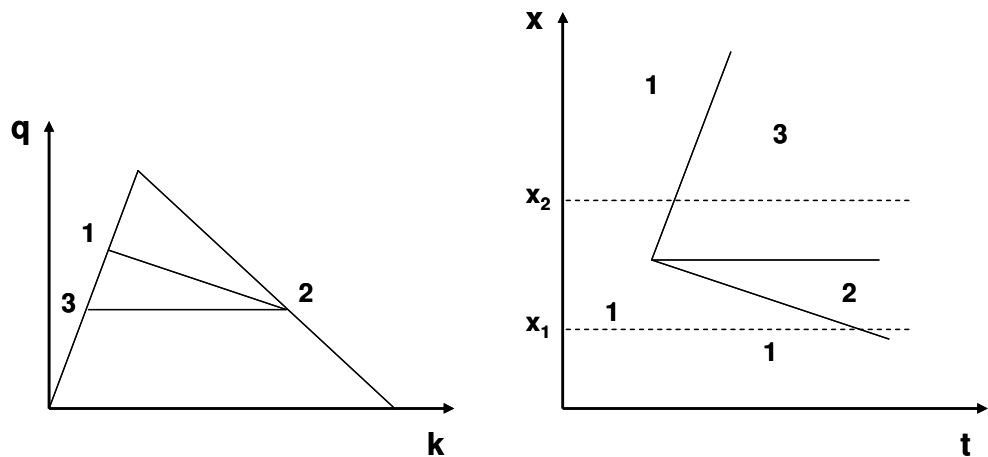
3.2.1. Vrije afwikkeling van het verkeer

Onder normale omstandigheden bevindt het verkeer zich in een vrije afwikkeling. De gemiddelde snelheden u_1 en dichtheden k_1 zijn dan kleiner dan respectievelijk de kritische snelheden u_c en kritische dichtheden k_c . Als gevolg van het incident zal er congestie ontstaan doordat de restcapaciteiten kleiner zijn dan de verkeersvraag.

Als we een situatie beschouwen waarin de verkeersvraag q_1 stationair (tijdsafhankelijk) is, er geen sprake van congestie is en we veronderstellen dat het fundamenteel diagram $q = Q(k)$ de verkeersafwikkeling goed beschrijft, kan met behulp van schokgolftheorie de verkeerscondities die ontstaan goed worden weergegeven (zie onderstaande figuur).

Figuur 3.4:

Verkeersafwikkeling volgens schokgolftheorie indien de verkeersvraag groter is dan de restcapaciteit en er voor het incident sprake is van een vrije afwikkeling



Rijbaanniveau

In bovenstaand figuur zijn de verkeerscondities te zien welke verwacht worden op rijbaanniveau. Stroomafwaarts van het incident zal zich toestand 3 voordoen en stroomopwaarts toestand 2. De schokgolf stroomopwaarts, die toestand 1 en 2 van elkaar scheidt, plant zich afhankelijk van de intensiteit uit met een snelheid van ongeveer 0 tot 20 km/h. Stroomafwaarts van het incident zal een intensiteitvacuüm (overgang tussen toestand 1 en 3) ontstaan dat zich voortplant met een snelheid gelijk aan de rijnsnelheid. Het verschijnsel dat de schokgolf stroomafwaarts zich veel sneller voortplant dan de schokgolf stroomopwaarts impliceert onder de hier geldende condities, dat in het

algemeen de gevolgen van een incident eerder stroomafwaarts dan stroomopwaarts waarneembaar zullen zijn. Dit is natuurlijk wel afhankelijk van de plaats van het incident ten opzichte van de plaats van de detectoren.

Stroomopwaarts van het incident zal dus na enige tijd congestie ontstaan. De gemiddelde intensiteiten die stroomopwaarts van het incident gemeten worden, na het plaatsvinden van het incident, zullen gelijk zijn aan de restcapaciteit. De gemiddelde snelheden zullen als gevolg van filevorming lager zijn dan normaal. Stroomafwaarts van het incident zal sprake zijn van een vrije afwikkeling. De gemeten intensiteiten zullen daar dan gelijk zijn aan de restcapaciteit. De snelheden stroomafwaarts van het incident zullen in ieder geval groter zijn dan de kritische snelheden. Indien de afstand tussen de eerste meetlocatie stroomafwaarts en het incident klein is, zullen de gemeten snelheden lager liggen omdat het verkeer nog niet volledig uitge accelereerd is.

Rijstrookniveau

Op rijstrookniveau zullen voornamelijk de verkeerscondities tussen de stroken verschillend zijn. Op de niet geblokkeerde rijstrook of rijstroken zal de intensiteit gelijk zijn aan de restcapaciteit. Op de geblokkeerde rijstrook of rijstroken zijn de intensiteiten dan nul. In vergelijking met de situatie voor het incident zal het aandeel verkeer op de niet geblokkeerde rijstroken dus toenemen.

Afhankelijk van de locatie van het incident in relatie tot de detector wordt er dus een verandering in de verdeling van het verkeer over de rijstroken verwacht. Voornamelijk in de buurt van het incident zullen de meeste voertuigen op de vrije stroken rijden, terwijl een enkel voertuig de congestie zal omzeilen door relatief lang op de geblokkeerde strook (of stroken) te blijven rijden. Stroomafwaarts van het incident zal het verkeer ook weer gebruik gaan maken van de geblokkeerde strook en zal de intensiteit op de deze strook groot zijn. Verder stroomopwaarts zal het verkeer zich weer gelijkmatiger over alle rijstroken verdelen en zal het verkeer zich over de rijstroken verdelen zoals in een normale situatie.

Als het incident de gehele weg afsluit zal er geen informatie te halen zijn uit het rijstrookgebruik als wordt gekeken naar de verschillen op één locatie. In dat geval moeten ook locaties stroomopwaarts en of stroomafwaarts in beschouwing worden genomen.

3.2.2. Congestie

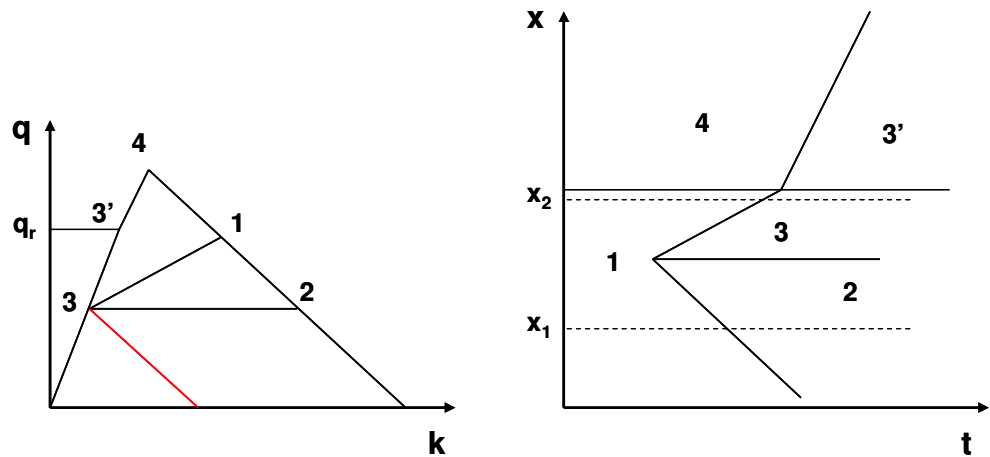
Als gevolg van bijvoorbeeld een toerit of een wegversmalling is het mogelijk dat er al voordat het incident heeft plaatsgevonden sprake is van congestie. In deze subparagraaf worden de te verwachten verkeerscondities in deze situatie op zowel rijbaanniveau als rijstrookniveau besproken.

Rijbaanniveau

Als we er weer vanuit gaan dat de verkeersvraag q_1 stationair (tijdsonafhankelijk) is en dat het fundamenteel diagram $q = Q(k)$ de verkeersafwikkeling goed beschrijft, kunnen de verkeerscondities met behulp van schokgolftheorie goed worden beschreven. In onderstaande figuur is deze situatie geschetst waarbij de verkeersvraag (van bijvoorbeeld een toerit) gelijk is aan q_r .

Figuur 3.5:

Verkeersafwikkeling volgens schokgolftheorie indien de verkeersvraag groter is dan de restcapaciteit en er voor het incident al sprake is van congestie



In principe is deze situatie vergelijkbaar met het geval dat er vrije afwikkeling van het verkeer zou zijn. Het grote verschil met een situatie van vrije afwikkeling is echter dat de schokgolf stroomafwaarts die de toestand 1 en 2 onderscheidt, zich met een aanmerkelijk kleinere snelheid voortplant dan de kritische snelheid. Dit zou dus kunnen betekenen dat de snelheid waarmee de schokgolf zich stroomafwaarts voortplant kleiner kan zijn de snelheid waarmee de schokgolf stroomopwaarts zich voortplant. De gevolgen van het incident kunnen hierdoor eerder de detector stroomopwaarts bereiken dan de detector stroomafwaarts. Dit is wel weer afhankelijk van de plaats van het incident ten opzichte van de detectoren.

Stroomafwaarts zal enige tijd na het incident een vrije afwikkeling gemeten worden. De gemiddelde snelheden zullen stroomafwaarts doorgaans groter zijn dan de kritische snelheden.

Rijstrookniveau

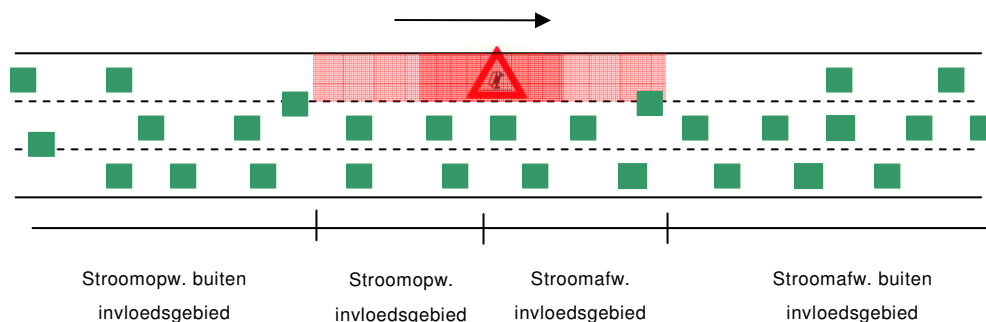
In het geval van 'normale' congestie zijn op alle rijstroken vergelijkbare verkeerscondities te verwachten. In het geval van een incident waarbij een of meerdere rijstroken afgesloten zijn, zal gedurende het incident het rijstrookgebruik tussen de stroken grote verschillen vertonen. Op de geblokkeerde rijstroken zijn de intensiteiten nul, terwijl het aandeel op de overige rijstroken toeneemt. Stroomafwaarts van het incident zal de restcapaciteit van de rijbaan weer groter zijn en zullen op alle rijstroken de intensiteiten gelijk zijn aan de restcapaciteit.

3.3 Verkeersafwikkeling als vraag kleiner dan restcapaciteit ⁷

In het geval dat de restcapaciteit na het incident nog groter is dan de vraag (zie figuur 3.6), zal het verkeer weinig hinder van het incident ondervinden.

Figuur 3.6:

Verkeersafwikkeling indien restcapaciteit na het incident **groter** is dan verkeersvraag



Rijbaanniveau

De intensiteiten stroomopwaarts zullen niet veranderen ten opzichte van de intensiteiten stroomafwaarts of van de situatie voor de blokkade. Het verkeer blijft immers in een vrije afwikkeling en past zijn snelheden niet of nauwelijks aan. Ook als wordt gekeken naar de gemiddelde snelheden zullen er dus geen duidelijke veranderingen zichtbaar zijn. Indien het incident dicht bij een detector heeft plaatsgevonden zal een lichte reductie in de gemeten snelheden gevonden kunnen worden. Als de restcapaciteit na het incident nog groter is dan de vraag zal op grond van de meetgegevens op rijbaanniveau dus geen eenduidige uitspraak kunnen worden gedaan of er sprake is van een incident of niet.

Rijstrookniveau

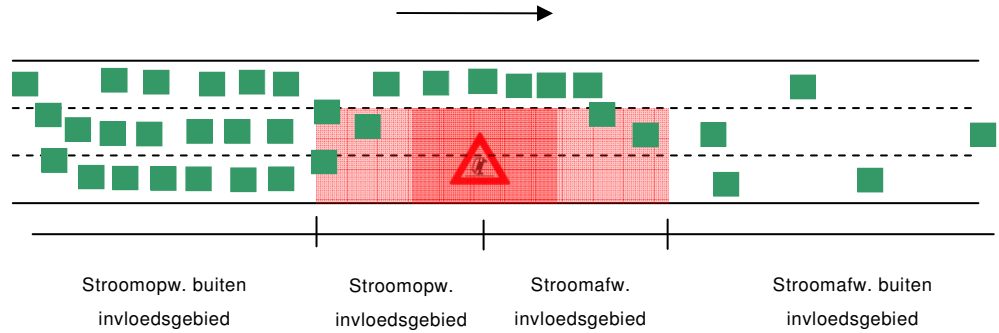
Op rijstrookniveau zullen in dit geval wel afwijkingen meetbaar zijn. Gelet op de intensiteiten zullen zowel stroomopwaarts als stroomafwaarts relatief lage intensiteiten op een of meerdere rijstroken worden gemeten. Het rijstrookgebruik op de locatie van het incident zal afwijkend zijn van een normale situatie. Op de niet geblokkeerde rijstroken zal het aandeel verkeer toenemen. Vooral in het geval dat de restcapaciteiten na het incident dus nog groter zijn dan de intensiteiten op dat moment zal het detecteren op rijstrookniveau iets toe kunnen voegen aan incident detectie.

3.4 Welke verkeerskenmerken zijn te verwachten?

In deze paragraaf worden de te verwachten kenmerken kort en overzichtelijk weergegeven. Hierbij wordt weer onderscheid gemaakt tussen de situatie waarbij de restcapaciteit na het incident kleiner is dan de verkeersvraag en wanneer de verkeersvraag kleiner is dan de restcapaciteit. Beide gevallen zijn in onderstaande figuren nogmaals weergegeven. Verder wordt onderscheid gemaakt tussen verkeerskenmerken die zich stroomopwaarts en stroomafwaarts en binnen of buiten het invloedsg gebied van het incident voordoen. In tabel 3.1 zijn de te verwachten verkeerskenmerken weergegeven.

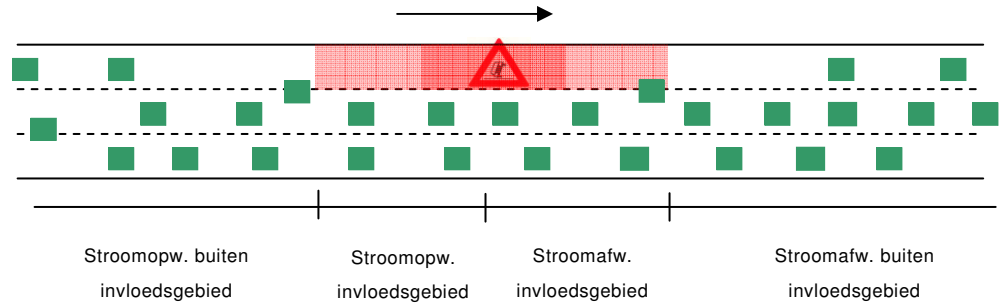
Figuur 3.7:

Verkeersafwikkeling indien restcapaciteit na het incident **kleiner** is dan verkeersvraag



Figuur 3.8:

Verkeersafwikkeling indien restcapaciteit na het incident **groter** is dan verkeersvraag



Tabel 3.1 Verkeerskenmerken die verwacht worden tijdens incidenten

	Intensiteit lager dan capaciteit weg	Intensiteit hoger dan capaciteit weg
Stroomopwaarts buiten invloedsgebied van blokkade	Op rijbaanniveau en rijstrookniveau zullen geen afwijkingen geconstateerd kunnen worden	lage snelheden op rijbaanniveau hoge dichtheden op rijbaanniveau rijbaanintensiteit gelijk aan restcapaciteit blokkade afwijkende strookverdeling (verkeer gelijkmatiger verdeeld over de rijstroken) afwijkende snelheidsverdeling (lage snelheden)
Stroomopwaarts binnen invloedsgebied blokkade (strookwisselingsgebied)	Licht afwijkende snelheden op rijbaanniveau normale dichtheden op rijbaanniveau normale rijbaanintensiteit afwijkende rijstrookverdeling (zeer weinig of geen verkeer op afgesloten rijstrook) iets afwijkende snelheidsverdeling	lage snelheden hoge dichtheden rijbaanintensiteit gelijk aan restcapaciteit blokkade afwijkende rijstrookverdeling (zeer weinig of geen verkeer op afgesloten rijstrook) afwijkende snelheidsverdeling (lage snelheden)

Stroomafwaarts binnen invloedsgebied blokkade (strookwisselingsgebied)	mogelijke licht afwijkende snelheden normale dichtheden normale rijbaanintensiteiten afwijkende rijstrookverdeling (weinig tot geen verkeer op afgesloten rijstrook) iets afwijkende snelheidsverdeling	kritische snelheden kritische dichtheden rijbaanintensiteit gelijk aan restcapaciteit blokkade afwijkende rijstrookverdeling (weinig tot geen verkeer op afgesloten rijstrook) afwijkende snelheidsverdeling (lage snelheden)
Stroomafwaarts buiten invloedsgebied blokkade	Op rijbaanniveau en rijstrookniveau zullen geen afwijkingen geconstateerd kunnen worden	hoge snelheden lage dichtheden rijbaanintensiteit gelijk aan restcapaciteit blokkade afwijkende rijstrookverdeling (verkeer meer verdeeld over de rijstroken) afwijkende snelheidsverdeling (hogere snelheden dan normaal)

4. Beschrijving Incident detectie algoritmen

In dit hoofdstuk wordt ingegaan op bestaande methoden om incidenten te detecteren. In de eerste twee paragrafen wordt kort ingegaan op de belangrijkste punten wat er in de literatuur te vinden is betreffende incident detectie en welke algoritmes er in de loop der jaren zoal ontwikkeld zijn. In de laatste drie paragrafen worden een drietal methoden nader bekeken.

4.1 Incident detectie in de literatuur

Incidenten zijn de onvoorspelbare en daardoor moeilijk te bestrijden oorzaken van files. Het tijdig detecteren van incidenten kan de gevolgen van het incident beperken. Om detecteren van incidenten te vergemakkelijken en versnellen zijn sinds 1970 tal van Automatische Incident Detectie (AID) algoritmes ontwikkeld. Wat voor soort detectiemethoden kunnen worden toegepast is afhankelijk van de beschikbare data (lusgegevens, videobeelden, individuele voertuiggegevens) In dit onderzoek wordt onderzoek gedaan naar incident detectie aan de hand van meetlusgegevens. Er wordt dan ook niet verder ingegaan op andere methoden.

Veel van de ontwikkelde methoden voor incident detectie methoden vanaf 1970 zijn ook nu nog altijd operationeel. Enkele bekende en veel toegepaste algoritmes zijn: het McMaster algoritme (Gall en Hall, 1989; Hall et al., 1993), het California algoritme (Payne and Tignor, 1978; Levin et al., 1979) en het Minnesota algoritme (Stephanedes et al., 1992; Stephanedes and Chassiakos, 1993). Al deze methode proberen incidenten te detecteren gebruik makend van lusdata.

De verschillende detectiemethoden verschillen alle in hun criteria voor de detectie, oftewel de verkeerskundige kenmerken waaraan het algoritme een incident herkent. Veel van de methoden hebben de overeenkomstige eigenschap dat ze niet incidenten op zich detecteren, maar congestievorming als gevolg van het incident, dat ze daarna controleren op de oorzaak van de congestie. Deze methoden moeten dus in staat zijn om congestie door incidenten te kunnen onderscheiden van congestie door bottlenecks en start-stop golven.

Om de prestaties van verschillende detectiemethoden te vergelijken worden over het algemeen drie parameters bekeken: het aantal gedetecteerde incidenten (detection rate), het aantal valse alarmen (false alarm rate) en de detectietijd (time to detect).

- *Het aantal gedetecteerde incidenten* is het aantal juist gedetecteerde incidenten in verhouding met het totaal aantal incidenten dat heeft plaatsgevonden. Een incident is juist gedetecteerd indien de melding van het incident ook is geconstateerd door bijvoorbeeld verkeerscentrales.
- Het aantal keer dat een incident detectie algoritmes een melding van een incident geeft, terwijl er geen sprake is van een incident wordt gesproken over het *aantal valse alarmen*
- De *detectietijd* is het verschil in tijd tussen het plaatsvinden van

een incident (of de meldtijd van een incident) en de tijd dat het incident door het algoritme wordt gedetecteerd

Deze drie parameters zijn onderling afhankelijk. Over het algemeen geldt dat door een toename in het aantal gedetecteerde incidenten ook het aantal valse alarmen toeneemt. Andersom geldt dat als het aantal valse alarmen verminderd wordt, het algoritme niet meer gevoelig genoeg is en het aantal gedetecteerde incidenten daalt. Een toename van de detectietijd, waardoor de data beter geanalyseerd kan worden, zorgt over het algemeen voor een verbetering in de andere twee parameters.

4.2 Indeling Incident Detectie algoritmes

De detectiemethoden die in de loop der jaren zijn ontwikkeld zijn door Martin et al. (2001) in vier hoofdtypes ingedeeld. Deze vier hoofdtypes zijn: (1) patroon herkenning (2) toepassing van chaos theorie, (3) statische methoden, (4) kunstmatige intelligentie. De laatste methode is pas sinds enkele jaren in ontwikkeling. Hieronder wordt elk hoofdtype kort toegelicht.

Patroon herkenning

De detectie methoden die tot deze groep behoren zijn het meest algemeen in gebruik. Ze maken gebruik van de gegevens van bezettingsgraad, intensiteiten en/of snelheden die over het algemeen wordt gemeten door detectielussen. Als er in de gemeten data patronen zichtbaar zijn die zich uitzonderen van de 'normale' situatie worden deze als mogelijke incidenten gezien. Vooraf ingestelde randvoorwaarden definiëren de normale situatie. Als de gemeten data niet voldoet aan deze randvoorwaarden, wordt een melding van een incident gegeven. Vooral het instellen van juiste randvoorwaarden is bij deze methoden vaak ingewikkeld en vergt veel tijd en deskundigheid. Het California algoritme (Payne and Tignor, 1978; Levin et al., 1979) is een voorbeeld van een detectie methode die tot dit type behoort. In Nederland is de blokkadedetector (Transpute, 2001), die berust op de werking van het California algoritme en het Presikhaaf algoritme (van Dommele) ontwikkeld die tot deze groep behoren.

Catastrophe theorie

Dit algoritme reageert op plotselinge verandering die zich voordoen in een van de drie variabelen uit het fundamenteel diagram (intensiteit, snelheid of bezettingsgraad), terwijl de overige gerelateerde variabelen zich normaal gedragen. In het geval van normale congestie zullen alle drie de variabelen namelijk geleidelijk veranderen, terwijl in het geval van een incident zich een uitzonderlijke situatie voordoet. In dat geval kan bijvoorbeeld alleen de snelheid plotseling afnemen, zonder een gerelateerde afname in intensiteit en bezettingsgraad. Het onderscheid tussen chaos theorie en model herkenning is dat de modelgebaseerde methoden gebruik maken van een individuele variabele waarbij vooraf drempelwaardes worden ingesteld, terwijl de chaos methode gebruik maakt van meerdere variabelen die worden vergeleken met situaties die zich in normale gevallen voordoen. Het McMaster algoritme (Persaud en Hal 1989) is een voorbeeld van dit type detectie methoden en is op veel plaatsen operationeel.

Statistische methoden

Statistische methoden vergelijken real time verkeersgegevens met verkeersvoorspellingen. Deze algoritmes modelleren het actuele verkeer en maken een voorspelling van de te verwachten meetwaarde. Als er onverwachte veranderingen in verkeer voorkomen die niet overeenkomen met de voorspelde waardes wordt dit als een incident geclassificeerd. Voorbeelden van algoritmes

in deze categorie zijn: High Occupancy (HIOCC) algorithm (Collins and Martin, 1979), Bayesian algorithm (Levin and Krause, 1978) en het Auto-Regressive Integrated Moving-Average time series (ARIMA) algorithm (Ahmed and Cook 1982).

Kunstmatige Intelligentie (Artificial Intelligence AI)

Incident detectie algoritmes van deze types is een meer recente ontwikkeling. Deze algoritme herkennen incidenten door ofwel een regelgebaseerde (rule-based) algoritme ofwel een algoritme dat 'geleerd' heeft om incidenten te herkennen. Voorbeelden van dit type zijn Neural Network (Stephanedes 1995) en GAID: Genetic Adaptive Incident Detection (Roy en Abdulhai, 2003).

Er bestaat niet alleen de mogelijkheid om detectiemethoden individueel in te zetten. Steeds meer nieuwe programma's en technologieën worden ontwikkeld waarin verschillende detectiemethoden aan een beslissingssysteem worden gekoppeld. Bij dit laatste geval werken dan bijvoorbeeld vijf algoritmes naast elkaar en het beslissingssysteem zorgt ervoor dat er alleen een incident melding wordt gegeven als bijvoorbeeld minimaal drie van de vijf methoden een incident gedetecteerd hebben (Sheu en Ritchy (2000), Levin et al. (1979) en Cohen en Ketselidou (1993)).

In onderstaande paragrafen worden een drietal bestaande algoritmen nader bekeken. Het eerste algoritme is de Blokkadedetector die al sinds enige tijd in Nederland in ontwikkeling is. Dit algoritme maakt gebruik van herkenbare patronen in de intensiteiten die herkenbaar zijn voor een incident en berust op de werking van het bekende California algoritme. Het tweede algoritme is het Presikhaaf Algoritme dat is ontwikkelt binnen Rijkswaterstaat. Ook dit model behoort tot het type patroon herkenning. Het verschil is echter dat dit algoritme voornamelijk zoekt naar herkenbare patronen in snelheden. Het derde algoritme is het McMaster Algoritme. Dit algoritme is sinds jaren succesvol operationeel in het Buitenland en in dit onderzoek wordt gekeken of er ook mogelijkheden zijn om dit algoritme in Nederland toe te passen.

4.3 De Blokkadedetector

De blokkadedetector (Transpute, 2002) is een incident detectie systeem dat ontwikkeld is in het kader van het MSS¹ project. Het principe van de Blokkadedetector is gebaseerd op het herkennen van typische patronen in de snelheid- en intensiteitsgegevens. De werking met het algoritme is te vergelijken met het California algoritme. Het grote verschil is dat de blokkadedetector gebruik maakt van intensiteit en snelheidsgegevens en het California algoritme van gegevens die in het buitenland worden gemeten door de detectielussen: de bezettingsgraad en de snelheid.

Uit evaluaties van de blokkadedetector is echter gebleken dat het percentage incidenten dat automatisch herkend wordt betrekkelijk laag is (ca 30%) en dat de detectietijd van de blokkadedetector aan de hoge kant is [evaluatie Blokkadedetector DHV, 2004]. In de volgende paragraaf wordt de werking van het algoritme beschreven.

¹ Het MSS project (Management Services Spitsstroken) is door RWS opgezet om landelijk over uniforme faciliteiten voor de bediening van spitsstroken (waar onder ook plusstroken en bufferstroken) te beschikken waarmee de verkeersleiding deze integraal kan openen, bewaken en sluiten.

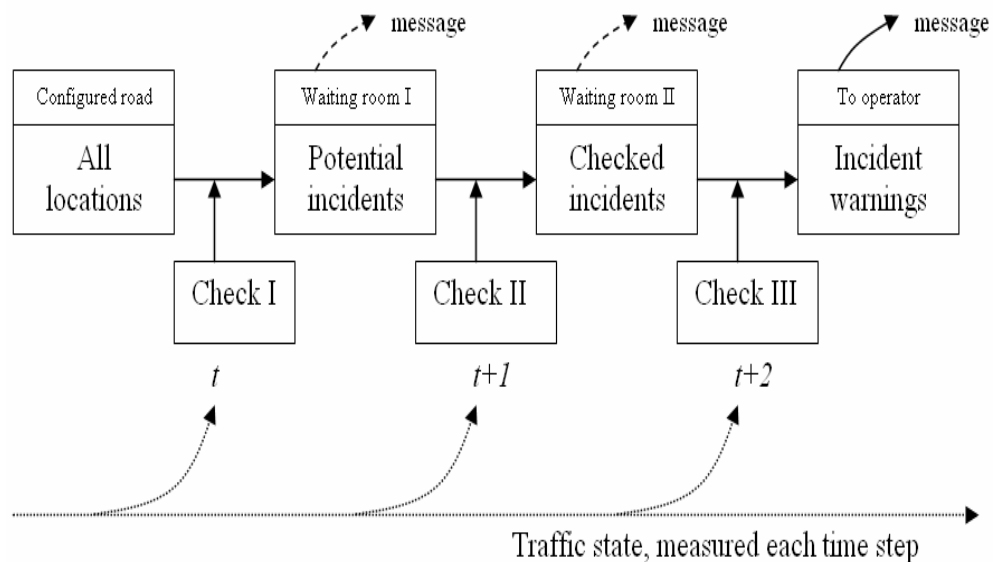
4.3.1. Het Algoritme

De blokkadedetector maakt voornamelijk gebruik van typische intensiteitskenmerken die zich voordoen tijdens een incident. In het vorige hoofdstuk is beschreven dat zowel binnen als buiten het invloedsgebied veranderingen in intensiteiten en snelheden te verwachten zijn indien de verkeersvraag hoger is dan de restcapaciteit. Als de intensiteit lager is dan de capaciteit worden er zowel binnen als buiten het invloedsgebied lage tot geen verschillen verwacht in intensiteiten en snelheden. Hieruit kan geconcludeerd worden dat verwacht wordt dat bij dit algoritme bij een volledige blokkade (een groot ongeluk) een goede detectie mogelijk is en bij een gedeeltelijke blokkade (bijv. als driekwart van het verkeer er nog langs kan) de detectie langer zal duren of het incident helemaal gedetecteerd zal worden.

De blokkadedetector controleert in een aantal achtereenvolgende minuten of de verkeersstroom voldoet aan bepaalde voorwaarden voor intensiteits- en snelheidspatronen die kenmerkend zijn voor een incident. In onderstaand figuur is het principe van het algoritme weergegeven.

Figuur 4.1:

De blokkadedetector bestaat uit drie checks die op achtereenvolgende minuten worden uitgevoerd (bron: Knibbe (2005)).



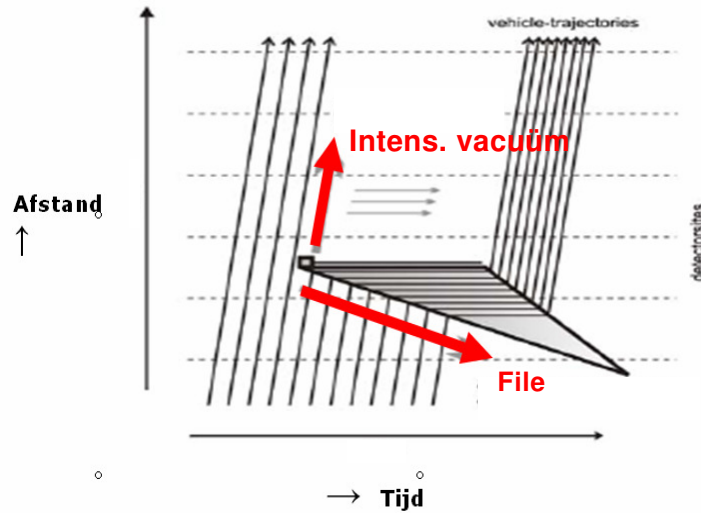
Zoals in de figuur te zien is bestaat het algoritme in totaal uit drie checks die op achtereenvolgende minuten worden uitgevoerd. Hierbij kan een incidentmelding worden gegeven na de eerste check, na het uitvoeren van twee checks of na het uitvoeren van alle drie de checks. In onderstaande subparagrafen worden de drie checks nader toegelicht.

Check 1: (tijdstip t)

De eerste check wordt voor elke minuut en voor elke meetlocatie uitgevoerd om plotselinge intensiteitsdalingen waar te nemen. Indien een 'groep verkeer' wordt gevolgd in de tijd is er onder normale incidentvrije verkeersomstandigheden weinig verschil te verwachten tussen intensiteiten van die groep. In hoofdstuk 2 heeft men al kunnen lezen dat in filevrije omstandigheden en bij intensiteit die hoger is dan de restcapaciteit, incidenten stroomafwaarts sneller te herkennen zijn dan stroomopwaarts (zie onderstaand figuur). Het algoritme zoekt dan ook in de eerste check naar een intensiteitsvacuüm stroomafwaarts en niet naar een filevorming.

Figuur 4.2:

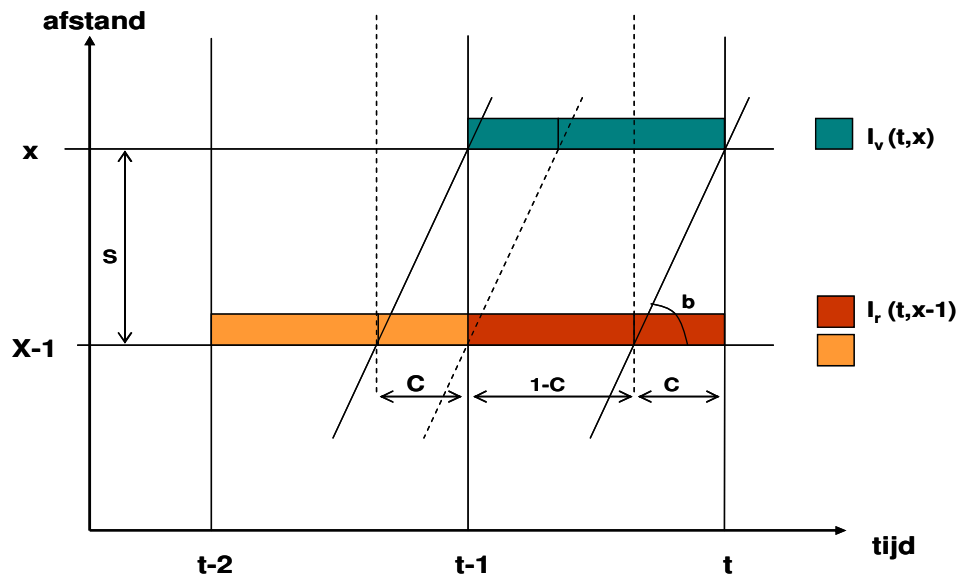
Een intensiteitvacuüm plant zich over het algemeen sneller voort dan de file.



Om een eventueel intensiteitsvacuüm te herkennen wordt voor elke minuut en locatie de gemeten intensiteit vergeleken met een te verwachten intensiteit op die locatie en dat tijdstip. Deze predictie van intensiteiten wordt gemaakt op basis van de metingen van de huidige en voorgaande minuten. Hiervoor wordt "een groep verkeer" terug in de tijd en plaats opgezocht en gekeken of de eigenschappen daar overeenkomen met de eigenschappen die de groep later heeft. In onderstaand figuur is dit weergegeven.

Figuur 4.3:

De voorspelling van de intensiteit is afgeleid uit de intensiteits- en snelheidsgegevens van een aantal meetlussen en tijdstappen terug (bron: RWS, 2005).



De formule voor het bepalen van de te verwachten intensiteiten (I_v) op een bepaald tijdstip ziet er als volgt uit:

$$I_v(t, x) = \underbrace{(1 - c(t, x)) \cdot I_r(t, x - 1)}_{\text{aandeel gegevens van locatie } x-1 \text{ op tijdstip } t} + \underbrace{c(t, x) \cdot I_r(t - 1, x - 1)}_{\text{aandeel gegevens van locatie } x-1 \text{ op tijdstip } t-1} \quad [4.1]$$

waarin: x = locatie meetlus
 $x-1$ = meetlus dichtstbijzijnde stroomopwaarts

I_r = relatieve intensiteit
 c = factor afh. van afstand en snelheid

De factor $c(t,x)$ is de reistijd in minuten tussen twee meetlussen. is dus afhankelijk van de afstand tussen de betreffende meetraai x en een meetraai stroomopwaarts $x-1$ en de locale snelheid $v(t,x)$.

$$c(t,x) = \max\left(60 \cdot \frac{d}{v(t,x)}, 1\right) \quad [4.2]$$

waarin: v = snelheid in km/h
 d = afstand tussen meetraai x en $x-1$ in km

Deze factor bepaalt het aandeel van de gegevens dat de groep verkeer in deze minuut op locatie $x-1$ bevond en het aandeel dat deze groep zich een minuut geleden op locatie $x-1$ bevond. Hoe hoger dus de snelheid is, hoe groter de fractie dat de vorige minuut invloed heeft. In dat geval wordt de voorspelling uitsluitend gebaseerd op de gegevens van de eerste meetlus stroomopwaarts op het betreffend tijdstip. Als de snelheid laag is nadert de factor c naar 1 en wordt rekening gehouden met de intensiteitsgegevens van de eerste meetlus stroomopwaarts.

Om systematische meetfouten van de detectielussen en invloeden van inhomogeniteiten zoals toe- en afritten te verkleinen wordt een relatieve intensiteit I_r berekend aan de hand van de gemeten intensiteiten (I_m) en de voortschrijdende gemiddelde intensiteiten I_s van twee locaties.

$$I_r(t,x) = \frac{I_s(t,x-1)}{I_s(t,x)} \cdot I_m(t,x) \quad [4.3]$$

waarin: $x+1$ = dichtstbijzijnde meetlus stroomafwaarts

Daarnaast wordt de intensiteit voortdurend afgevlakt door middel van onderstaande formule.

$$I_s(t,x) = A \cdot I_m(t,x) + (1-A) \cdot I_s(t-1,x) \quad [4.4]$$

waarin: $A = 0,3$ als $I_s(t) > I_s(t-1)$ ($A_{stijgend}$)
 $A = 0,4$ als $I_s(t) < I_s(t-1)$ (A_{dalend})

De voorspelde intensiteiten I_v worden vergeleken met de gemeten intensiteiten op dat tijdstip en de betreffende meetlus. Onder normale omstandigheden, wanneer er sprake is van een vrije afwikkeling, kan ervan worden uitgegaan dat de gemeten intensiteiten overeenkomen met de voorspelde intensiteiten. Indien deze verhouding beneden een bepaalde drempelwaarde komt (in hoofdstuk 7 wordt verder ingegaan op de waardes van de parameters), wordt dit punt gezien als een potentiële blokkade. In het geval dat de gemeten intensiteiten zeer veel kleiner is dan de verwachte intensiteiten kan er met *zekerheid* worden vastgesteld dat er zich een intensiteitsvacuüm voordoet. Indien 4.5 geldt wordt een melding aan de operator gegeven van een incident.

$$\frac{I_m(t, x)}{I_v(t, x)} < H_{q_klein} \quad [4.5]$$

Als de verhouding tussen de gemeten intensiteit en voorspelde intensiteit kleiner is dan een drempelwaarde, dan is er sprake van een verstoring in de verkeerstromen en een *mogelijk* incident. Er moet dan gelden:

$$\frac{I_m(t, x)}{I_v(t, x)} < H_{q_groot} \quad [4.6]$$

Indien aan deze voorwaarde wordt voldaan, wordt deze locatie in wachtkamer 1 gezet als een potentiële incident en wordt overgegaan naar check 2.

Check 2 (tijdstip t+1)

Check 2 wordt uitgevoerd op alle mogelijke incidenten die in check 1 zijn vastgesteld en in wachtkamer 1 zijn gezet. In deze stap wordt gecontroleerd of het ontstane intensiteitsvacuüm nog steeds aanwezig is en of de snelheid stroomafwaarts hoog genoeg is.

Indien zich op een locatie stroomafwaarts een groot intensiteitsval voordoet kan in de tweede stap met zekerheid worden vastgesteld dat het om een incident gaat en wordt een melding van een incident aan de operator doorgegeven. Check 3 hoeft dan niet meer te worden uitgevoerd. Er moet dan gelden:

$$\frac{I_m(t, x+1)}{I_v(t-1, x+1)} < G_{q_klein} \quad [4.7]$$

Als bovenstaande niet geldt worden onderstaande checks uitgevoerd om na te gaan er nog steeds een intensiteitsvacuüm aanwezig is:

- $\frac{I_m(t+1, x)}{I_v(t+1, x)} < H_{q_groot} \quad [4.8]$

- $\frac{I_m(t, x-1)}{I_v(t, x-1)} < H_{q_groot} \quad [4.9]$

- $\frac{I_m(t+1, x-1)}{I_v(t+1, x-1)} < H_{q_groot} \quad [4.10]$

Als minimaal 1 van bovenstaande condities geldt wordt vervolgens gekeken of de snelheid stroomafwaarts hoog genoeg is.

$$v(t+1, x+1) > V_{checken} \quad [4.11]$$

Indien minimaal een van de condities 4.8, 4.9 of 4.10 geldt en conditie 4.11 ook, is in deze stap een mogelijke blokkade gedetecteerd en wordt check drie daarna uitgevoerd voor deze potentiële incidenten die in wachtkamer 2 zijn

gezet.

Check 3 (tijdstip $t+2$)

In de laatste check worden de incidenten die in check 2 in de wachtkamer zijn gezet gecontroleerd of er een voldoende hoge snelheids- en intensiteitsverschil meetbaar is:

$$\frac{v(t+2, x-1)}{v(t+2, x+1)} < F_{v_melden} \quad [4.12]$$

$$\frac{I_m(t+2, x+1)}{I_s(t-1, x+1)} < F_{q_melden} \quad [4.13]$$

Indien beide voorwaarden voldoen wordt er een melding van een incident weergegeven.

4.4 Presikhaaf algoritme

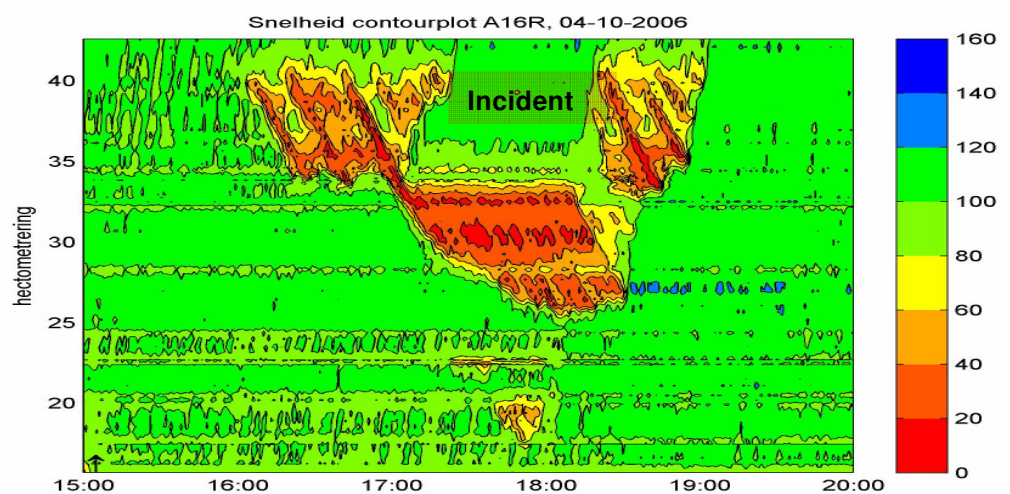
Het Presikhaaf algoritme (Van Dommele, Rijkswaterstaat te Presikhaaf) is ontwikkeld als alternatief op de blokkadedetector. De achterliggende gedachte van dit algoritme is dan ook berust op de werking van de blokkadedetector met als verschil dat dit algoritme voornamelijk kijkt naar snelheidsverschillen tussen locaties op verschillende tijdstippen in plaats van intensiteitverschillen. In de volgende paragraaf wordt de werking van deze methode besproken.

4.4.1. De werking van het algoritme

Snelheden zijn een van de karakteristieke meetwaarden waaraan een incident herkend kan worden. Het Presikhaaf algoritme zoekt naar snelheidsverschillen tussen opeenvolgende minuten en locaties. In het vorige hoofdstuk is beschreven dat bij incidenten de snelheden stroomopwaarts laag zullen zijn en stroomafwaarts kritisch en hoog. In onderstaand figuur is nogmaals een snelheid contourplot te zien van een incident, waarin duidelijk zichtbaar is dat er grote snelheidsverschillen waar te nemen bij incidenten.

Figuur 4.4:

Snelheidscontourplot van een incident.



Net als de blokkadedetector bestaat het Presikhaaf algoritme uit drie stappen, die op achtereenvolgende minuten en voor elke locatie wordt uitgevoerd, om voldoende zekerheid te scheppen of er een incident heeft plaatsgevonden. Zoals de blokkadedetector gebruik maakt van het ‘volgen van een groep verkeer’ om verwachte intensiteiten te bepalen, maakt dit algoritme gebruik van voortschrijdende gemiddelden om te bepalen of onverwachte veranderingen te constateren zijn.

Voortschrijdend gemiddelde

Het verkeer op de weg gedraagt zich niet als een ideale verkeerstroom, maar is juist erg grillig. Dit houdt in dat tussen gegevens van opeenvolgende minuten nogal verschillen kunnen voorkomen. Daarnaast bevinden zich in de data ook regelmatig afwijkingen. Om deze fouten te verkleinen wordt in het Presikhaaf algoritme niet twee opeenvolgende minuten met elkaar vergeleken, maar wordt de huidige minuut vergeleken met het voortschrijdend gemiddeld van de vijf voorgaande minuten. Onderstaand wordt ingegaan op de drie stappen van dit algoritme.

Stap 1:

In de eerste stap wordt voor elke minuut en locatie de gemeten snelheid vergeleken met het voortschrijdend gemiddelde van de laatste vijf minuten. Als er een snelheid wordt gemeten die veel lager is dan de minuten ervoor is er mogelijk iets aan de hand. Er moet dan gelden:

$$\frac{v(t, x)}{v_{vsg}(t, x)} < stap_1.1 \quad [4.13]$$

Een verlaging van de snelheid kan echter ook door een file zijn veroorzaakt. Om dit te controleren wordt gecontroleerd of de actuele snelheid stroomafwaarts hoog genoeg is:

$$v(t, x + 1) > stap_1.2 \quad [4.14]$$

Een snelheidsverlaging kan ook door een schokgolf veroorzaakt zijn. Om dit uit te sluiten wordt gecontroleerd of onderstaande voorwaarde geldt:

$$v_{vsg}(t, x + 1) < stap_1.3 \quad [4.15]$$

Als aan bovenstaande voorwaarden wordt voldaan is er een mogelijke blokkade en wordt stap 2 uitgevoerd voor deze potentiële incidenten.

Stap 2:

Deze stap wordt een minuut later uitgevoerd (op t+1) voor alle mogelijke incidenten die in de eerste stap zijn bepaald. Er wordt gecontroleerd of er nog steeds sprake is van een daling van de snelheid ten opzichte van het voortschrijdend gemiddelde en of de snelheid stroomafwaarts hoger is dan een drempel om een file uit te sluiten. Onderstaande condities moeten dan beide gelden:

$$\frac{v(t, x)}{v_{vsg}(t, x)} < stap_2.1 \quad [4.16]$$

$$v_{vsg}(t, x+1) < stap_2.2 \quad [4.17]$$

Ook wordt in deze stap gecontroleerd of de intensiteit stroomafwaarts lager is dan het voortschrijdend gemiddelde:

$$\frac{q(t, x+1)}{q_{vsg}(t, x+1)} < stap_2.3 \quad [4.18]$$

Als aan alle drie de voorwaarden wordt voldaan wordt deze locatie als mogelijk incident gezien en wordt in de volgende minuut de laatste check uitgevoerd.

Stap 3:

In de laatste stap wordt nogmaals gecontroleerd of de huidige snelheid lager is dan een grenswaarde en of er nog steeds een verschil in snelheid is tussen de actuele snelheid en de voortschrijdend gemiddelde snelheid.

$$v(t, x) < stap_3.1 \quad [4.19]$$

$$\frac{v(t, x)}{v_{vsg}(t, x)} < stap_3.2 \quad [4.20]$$

Als ook aan deze voorwaarden wordt voldaan, wordt een melding van een incident gegeven.

4.5 McMaster Algoritme

Het McMaster algoritme is sinds jaren succesvol operationeel in verkeersmanagement systemen over heel de wereld. Dit algoritme is ontwikkeld door de Traffic Research Group van de McMaster University van Toronto (Gall en Hall, 1989). Het algoritme bestaat uit twee delen. Ten eerste zoekt het algoritme naar congestievorming en vervolgens wordt bepaald of deze congestie is ontstaan door een incident of niet.

4.5.1. Bezettingsgraad

Het McMaster algoritme maakt gebruik van twee verkeersgegevens die beide worden gemeten worden door een enkele detectielus: de bezettingsgraad en de intensiteit. In Nederland bevinden zich detectielusparen en wordt de snelheid en intensiteit gemeten en zijn gegevens over de bezettingsgraad niet beschikbaar. Deze is met enkele aannames wel af te leiden van de intensiteit en snelheid. Voor een uitgebreidere afleiding van de bezettingsgraad wordt verwezen naar bijlage B.

De bezettingsgraad is een grootheid die staat voor de fractie van tijd dat een voertuig de detector *bezet*. Een voertuig dat een loop passeert "bezet" deze vanaf het moment dat de voorzijde van het voertuig aan het begin van de loop is totdat zijn achterkant aan het eind van de loop is. Voor elke voertuig bestaat een individuele bezettingsgraad periode.

Voor een periode T, waarin n voertuigen passeren, kan de bezettingsgraad β gedefinieerd worden als (Hoogendoorn et al., 2005):

$$\beta = \frac{1}{T} \sum_{i=1}^n b_i \quad [4.21]$$

waarin: b_i = individuele bezettingsgraad
 T = periode
 n = aantal gepasseerde voertuigen in periode T

Voor elke individueel voertuig is zijn bezettingsgraad afhankelijk van de snelheid die het voertuig op dat moment heeft en de lengte van dat voertuig plus de lengte van detector zelf.

Voor een individueel voertuig geldt:

$$b_i = \frac{L_i + L_{lus}}{u_i} \quad [4.22]$$

waarin: L_i = Lengte voertuig
 L_{lus} = Lengte detectielus
 u_i = snelheid voertuig

Indien ervan uit wordt gegaan dat de voertuigen een uniforme lengte en snelheid hebben geldt (Hall, 1992):

$$\beta = L_{tot} \frac{q}{u} \quad [4.23]$$

waarin: q = intensiteit (vtg/uur)
 L_{tot} = lengte detectielus + gem. lengte voertuigen (m)
 u = gemiddelde momentane snelheid (km/uur)

Bovendien geldt dat $q=k \cdot u$ en dus ook

$$\beta = L_{tot} \cdot k \quad [4.24]$$

waarin: k = dichtheid (vtg/km)

In werkelijkheid bestaat het verkeer niet uit een uniforme stroom van voertuigen. Wel kan gebruik worden gemaakt van een gemiddelde lengte van alle voertuigen. Hierbij wordt ervan uitgegaan dat:

- 70% personenauto's zijn met een lengte van gemiddelde lengte van 5 meter,
- 20% middellang verkeer is met een gemiddelde lengte van 6,5 meter
- 10% vrachtverkeer is met een gemiddelde lengte van 10 meter.

Dit geeft een gemiddelde lengte van alle voertuigen van 7,75 meter. De lengte van een detectielus bedraagt in Nederland 1,5 meter. De waarden voor de snelheid en intensiteit zijn beide bekend uit de verkeersgegevens, waarmee de bezettingsgraad benaderd kan worden.

4.5.2. Het algoritme

Het McMaster algoritme bestaat uit twee delen. Het eerste deel classificeert het verkeer in een van de vier mogelijke verkeerstoestanden op basis van de bezettingsgraad en de intensiteit die de verkeersstroom beschrijven. Als het

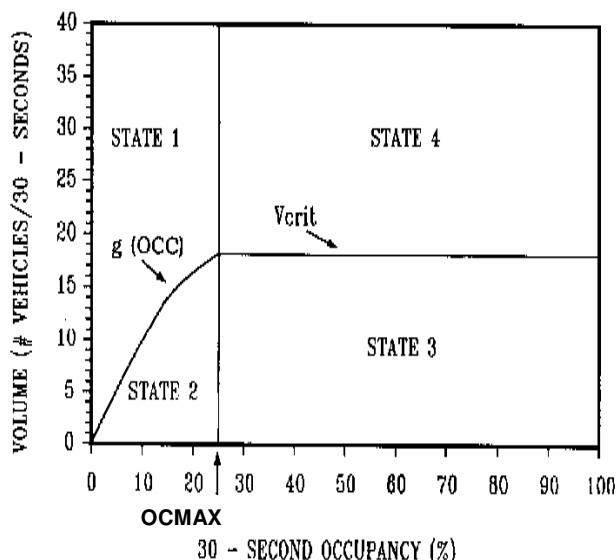
ware wordt gezocht naar meetwaarden die afwijken van het fundamenteel diagram. Het tweede deel gebruikt deze informatie om te bepalen of de congestie is ontstaan door een incident of niet.

Fase 1:

Om de verkeerstoestand te bepalen wordt het bezettingsgraad intensiteitsdiagram in vier delen verdeeld (zie figuur 4.5).

Figuur 4.5:

Geschematiseerde intensiteiten- dichtheids grafiek om de verkeersklassen te definiëren. (Bron: TRR 1232, 1989)



state 1: geen congestie
state 2: congestie
state 3: congestie
state 4: accelererend verkeer dat uit file komt

Vertikaal wordt de grafiek in tweeën gedeeld door de lijn OCMAX. OCMAX is de maximale bezettingsgraad waarbij nog geen congestie ontstaat. Aan de zijde van de grafiek waar de bezettingsgraad lager is dan de maximale bezettingsgraad wordt de grafiek verdeeld in toestand 1 en 2. Deze toestanden worden gescheiden door de lijn $g(occ)$. $g(occ)$ is een lijn die de plot benadert van de gemeten bezettingsgraad en de intensiteit, dit is dus het deel van het fundamenteel diagram waarin sprake is van vrije afwikkeling. Rechts van de lijn OCMAX verdeelt de lijn V_{crit} , de kritische intensiteit, de grafiek in verkeerstoestand 3 en 4. In de volgende paragraaf wordt verder ingegaan op de gebruikte parameters.

In totaal zijn er dus vier verkeerstoestanden te onderscheiden:

- State 1 is gelokaliseerd aan de bovenzijde van $g(occ)$ en heeft een bezettingsgraad die kleiner is dan OCMAX. Indien een meetgegeven in dit deel valt is er sprake van een normale situatie.
- State 2 beschrijft de verkeerscondities stroomafwaarts van de congestie. In dat geval is de bezettingsgraad kleiner dan OCMAX en de gemeten intensiteit ligt lager dan $g(occ)$
- State 3 is een tweede vorm van congestie waar de bezettingsgraad groter is dan de maximale bezettingsgraad en de intensiteit lager dan de kritische intensiteit.
- State 4 geeft de verkeerssituatie stroomafwaarts van een blokkade weer met accelererende snelheden weer, waar de bezettingsgraad groter is dan OCMAX en de intensiteiten groter dan de kritische intensiteit.

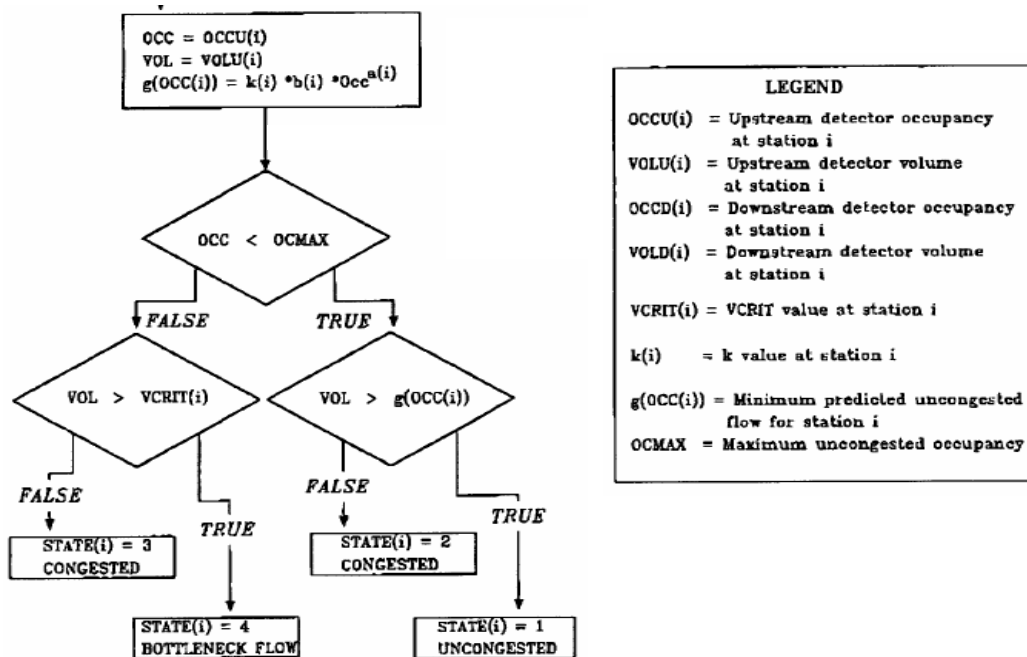
Als op een bepaalde meetlocatie toestand 2 of 3 wordt geconstateerd is er sprake van congestie en wordt overgegaan naar fase 2 om te bepalen of deze congestie is ontstaan door een incident of niet.

Fase 2:

In deze fase wordt geanalyseerd of de gedetecteerde file op een bepaalde meetlocatie x veroorzaakt is van een incident of dat er sprake is van 'normale' filevorming. Om dit te kunnen bepalen wordt voor de eerst volgende meetlocatie stroomafwaarts ($x+1$) de verkeerstoestand bepaald. Als op de meetlocatie stroomafwaarts zich verkeerstoestand 1 of 2 voordoet is een incident de oorzaak van de congestie. In figuur 4.6 is fase 2 overzichtelijk weergegeven.

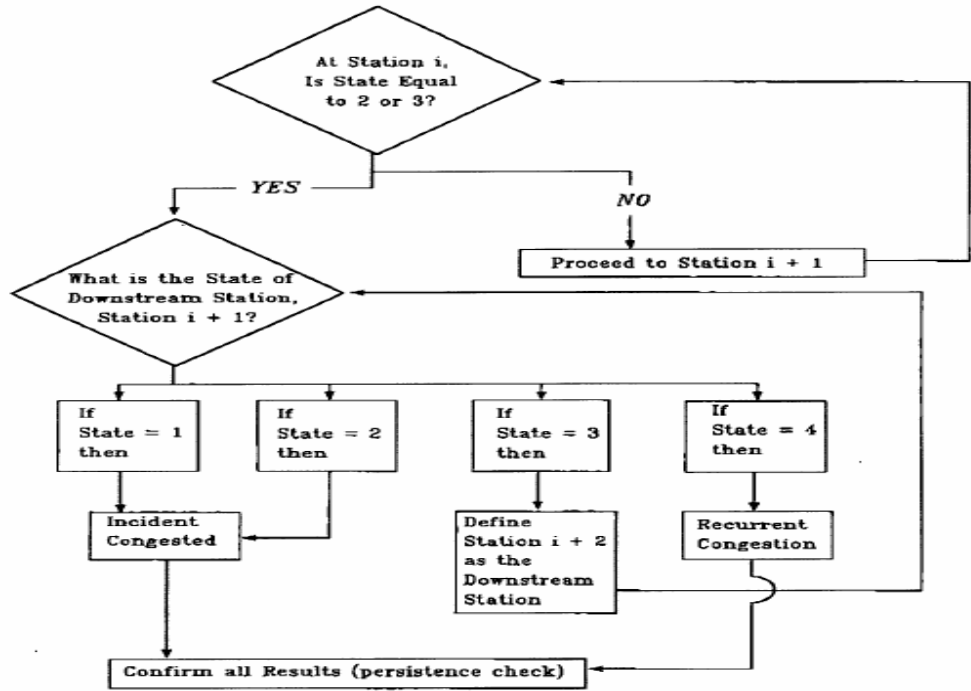
Figuur 4.6:

Beslissingsboom voor fase 1 van het McMaster algoritme. (Bron: TRR 1232, 1989)



Figuur 4.7:

Beslissingsboom voor fase 1 van het McMaster algoritme (Bron: TRR 1232, 1989)



Hoe het verkeer zich gedraagt en in welke verkeerstoestand het zich bevindt is afhankelijk van lokale omstandigheden, zoals het aantal rijbanen en of er op- of afritten in de buurt zijn. Door welke waarden de vier verkeerstoestanden kunnen worden onderscheiden hangt dus af van de meetlocatie en moet voor elke locatie apart worden bepaald. Hieronder worden de parameters die de vier verkeerstoestanden onderscheiden nader toegelicht.

g(occ)

Als alle meetwaarden van de intensiteiten worden uitgezet tegen de bezettingsgraad vormen de punten nagenoeg het congestievrije deel van het fundamenteel diagram die benaderd kan worden met onderstaande formule:

$$f(occ) = b \cdot occ^a \tag{4.25}$$

waarin: *f(occ)* = intensiteit
occ = bezettingsgraad
a, b = locatieafhankelijke parameters

De waardes *a* en *b* zijn locatieafhankelijk en worden in Matlab voor elke locatie afzonderlijk bepaald.

Verkeerstoestand 1 en 2 wordt onderscheiden door onderstaande formule:

$$g(occ) = k \cdot f(occ) \tag{4.26}$$

Hierin is *k* een waarde tussen 0 en 1 die zorgt voor een verschuiving van de lijn *f(occ)* naar beneden. In de beschrijving van Gall en Hall (1989) wordt voor *k* ongeveer de waarde 0,95 aangegeven. Aangezien de waarde van de bezettingsgraad hier wordt berekend waarbij wordt uitgegaan van een homogene verkeerstrom, zullen de afwijkingen wat groter zijn. De waarde *k* zal dan ook groter aangenomen moeten worden en zal tussen 0,7 en 0,9 liggen.

Vcrit

Verkeerstoestand 3 en 4 wordt onderscheiden door de kritische intensiteit, Vcrit. Deze waarde kan voor elke locatie online bepaald worden aan de hand van historische data van incidenten en snelheden.

OCMAX

De waarde van OCMAX kan berekend worden de kritische intensiteit in formule 4.25 in te vullen.

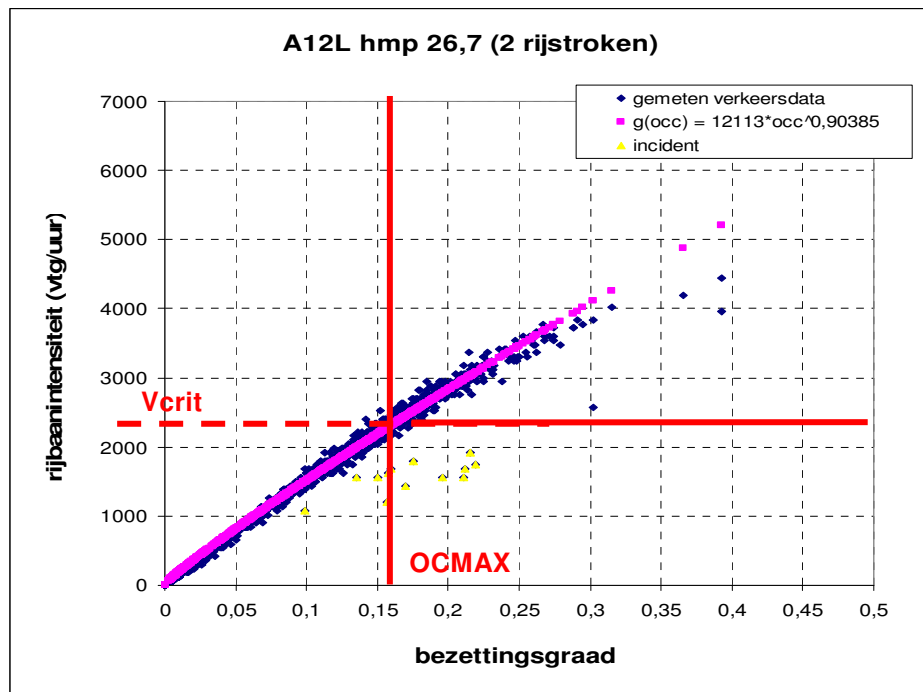
Voor een incident op de A12 op hmp 26,7 is in figuur 4.8 de bezettingsgraad tegen de rijbaanintensiteit geplot als voorbeeld voor dit algoritme. De meetwaarden zijn te benaderen met de lijn:

$$g(\text{occ}) = 12113 * \text{occ}^{0,90385}$$

In figuur 4.8 geven de gele punten de meetwaarden aan ten tijde van het incident. Deze wijken duidelijk af van de 'normale' situatie. In dit geval zal het incident dus door het McMaster algoritme worden gedetecteerd.

Figuur 4.8:

Plot van de bezettingsgraad uitgezet tegen de rijbaanintensiteiten voor de A12, hmp 26,7



5. Onderzoek bruikbaarheid rijstrookdata

In dit hoofdstuk wordt onderzocht of verkeersdata op strookniveau nuttig kunnen zijn voor het detecteren van incidenten. In hoofdstuk 3 is aangegeven dat voornamelijk in het rijstrookgebruik veranderingen te verwachten zijn tijdens incidenten. In dit hoofdstuk wordt onderzocht of rijstrookverdelingen een indicator kunnen zijn voor het detecteren van incidenten. Voor een aantal incidenten wordt de verkeersdata geanalyseerd. In de eerste twee paragrafen wordt ingegaan op de onderzochte incidenten en de analysemethode. In paragraaf 5.3 wordt onderzocht of en welke afwijkingen tijdens incidenten te vinden zijn en in paragraaf 5.4 wordt gekeken of deze afwijkingen uitzonderlijk genoeg zijn om incidenten te herkennen. In de laatste paragraaf wordt gekeken of onderscheid in situaties met vrije afwikkeling en congestie noodzakelijk is.

5.1 De onderzochte incidenten

Voor het onderzoek naar strookverdelingen zijn een drietal incidenten nader bekeken. De melding van deze incidenten komt van de gegevens van VCNL. Bij het selecteren is getracht om variatie te hebben in de incidenten: verschillende wegen en richtingen, verschillende locaties, rijdstippen en dagen, verschillende oorzaken en file-zwaartes. In onderstaande tabel zijn de onderzochte incidenten weergegeven.

Tabel 5.1 De incidenten die worden geanalyseerd

locatie	hmp	datum	tijd melding	Week dag	File lengte (km)	File duur (min)	File zwaarte (km*min)	omschrijving
A12 Links	26,7	23 juni 2006	12:33 - 12:57	vrijdag	2,7	25	68	Verkeers-ongeval
A12 Rechts	20,5	20 juni 2006	15:50 - 16:06	dinsdag	2	17	34	Gestrand voertuig
A20 Rechts	31,5	26 juli 2006	17:07 - 18:31	dinsdag	4,5	85	380	Verkeers-ongeval

5.2 Analysemethode

In deze analyse wordt de verdeling van het verkeer over de rijstroken onderzocht. Hoe het verkeer zich verdeelt over de rijstroken is afhankelijk van de locatie en de rijbaanintensiteit. Verwacht wordt dat het verkeer ten tijde van een incident zich anders over de beschikbare rijstroken verdeelt dan in normale incident vrije situaties. In eerste instantie wordt voor elk incident onderzocht of

zich ten tijde van het incident herkenbare afwijkende patronen vindbaar zijn (zijn incidenten hieraan te herkennen?). Daarnaast wordt onderzocht of deze afwijkende rijstrookverdelingen zich voldoende onderscheiden van de reguliere situatie (is die afwijking tijdens incidenten uitzonderlijk genoeg?). Indien dit niet het geval zou zijn, zouden het aantal valse meldingen veel te hoog worden.

5.3 Verdeling verkeer over rijstroken

De verdeling van het verkeer over de rijstroken zal afhankelijk zijn van de rijbaanintensiteit. Bij lage intensiteiten zal vrijwel alle verkeer gebruik maken van de rechter rijstrook. Als de intensiteiten toenemen, wordt het aandeel verkeer op de linker rijstroken geleidelijk hoger. In deze paragraaf wordt onderzocht of tijdens incidenten een herkenbare afwijking in de rijstrookverdeling te vinden is. Dit wordt gedaan voor de incidenten die eerder in het hoofdstuk zijn benoemd. Voor deze analyse is gebruik gemaakt van de minuutgegevens van de intensiteiten per rijstrook. Het doel hiervan is te bepalen of er karakteristieken te vinden zijn in de verdeling van het verkeer over de rijstroken tijdens incidenten.

Incident 1: A12L hmp 26,700

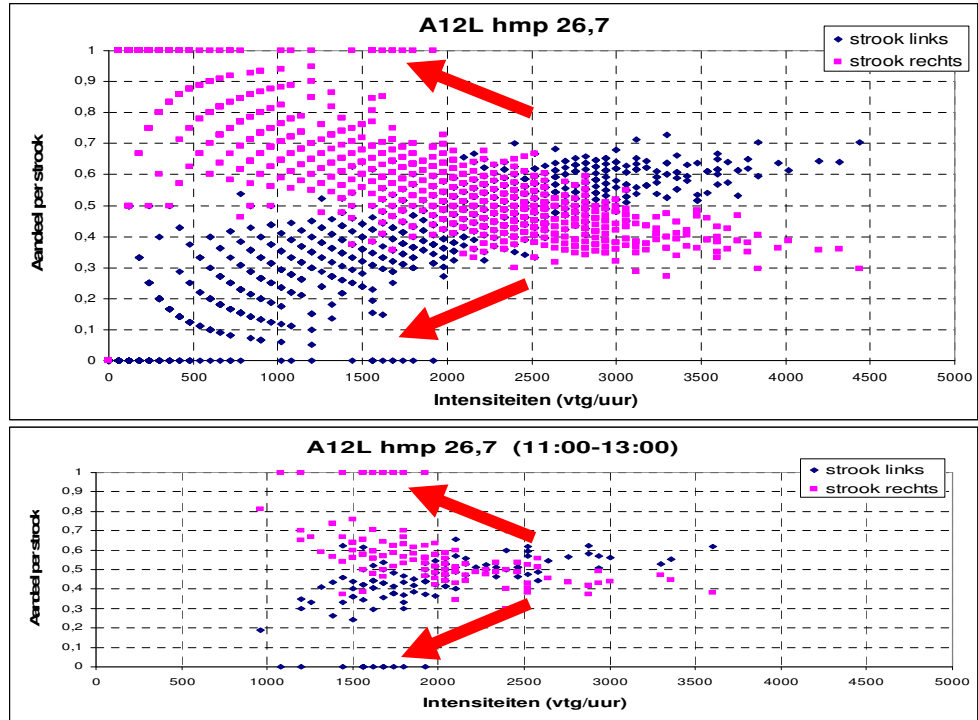
Op deze locatie bestaat de rijbaan uit twee stroken. In onderstaand figuur is een plot te zien van het aandeel van het verkeer per strook uitgezet tegen de rijbaanintensiteit over de gehele dag op de meetlocatie waar het incident zich heeft voorgedaan. Op de horizontale as staan de rijbaanintensiteiten en op de verticale as het aandeel verkeer per rijstrook. Er is een duidelijk patroon zichtbaar in de verdeling van het verkeer:

- Bij lage intensiteiten (0-1500 vtg/uur) rijdt het meeste verkeer op de rechter rijstrook (50-100%).
- Als de intensiteiten toenemen is wordt het aandeel op de linker rijstrook steeds groter
- Bij hoge intensiteiten (vanaf ongeveer 2500 vtg/uur) is het aandeel verkeer op de linker rijstrook hoger dan op de rechter.

Bij rijbaanintensiteiten tussen de 1500 en 2000 vtg/uur (rode pijlen) is een afwijking in dit patroon zichtbaar. In de onderste grafiek zijn de gegevens geplotted van de periode dat het incident zicht heeft voorgedaan. Hieruit kan men opmaken dat deze uitzonderlijke verdeling zich heeft voorgedaan ten tijde van het incident.

Figuur 5.1:

Aandeel verkeer per rijstrook per intensiteit klasse: bovenste figuur voor de gedurende de gehele dag en onderste figuur in de periode van het incident

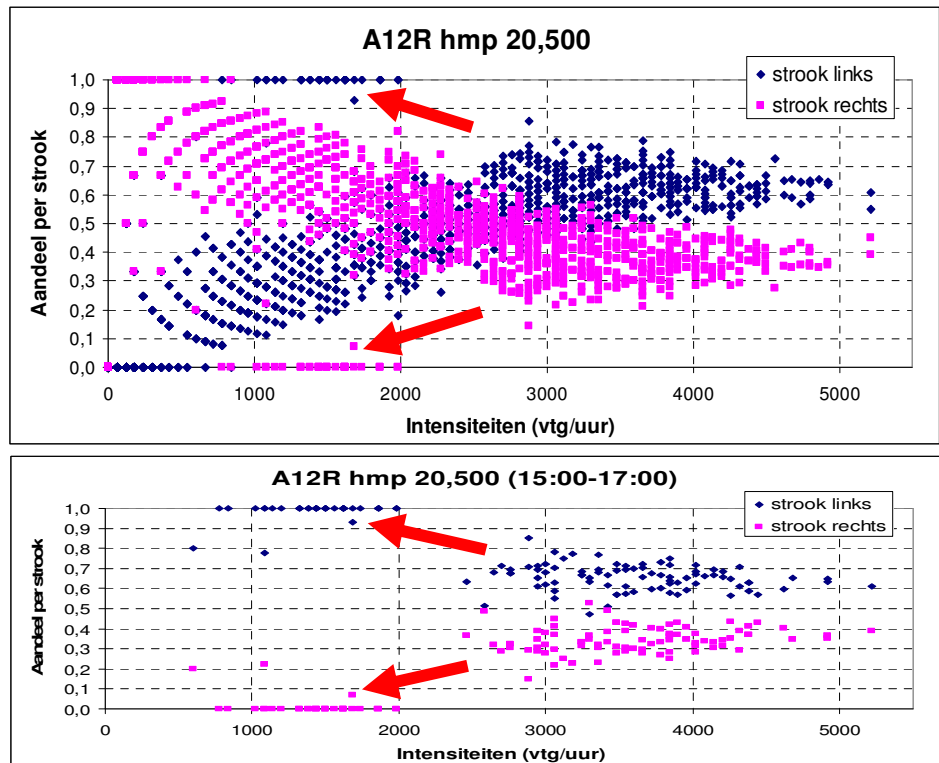


Incident 2: A12R hmp 20,500

De A12 bij hmp 20,500 bestaat uit twee rijstroken. Bij dit incident is een gestrand voertuig de oorzaak. Ook hier is weer (hetzelfde als bij het vorig incident) een duidelijk herkenbaar patroon zichtbaar in de verdeling van de rijstroken. Bij de rode pijlen is een duidelijke afwijking zichtbaar in dit patroon. In de onderste figuur is een plot van de meetgegevens in de periode dat het incident heeft plaatsgevonden. Hieruit kan men opmaken dat deze uitzonderlijke meetwaarden zich ook hier hebben voorgedaan ten tijde van het incident.

Figuur 5.2:

Aandeel verkeer per rijstrook per intensiteit klasse: bovenste figuur voor de gedurende de gehele dag en onderste figuur in de periode van het incident



Incident 3: A20R hmp 31,550

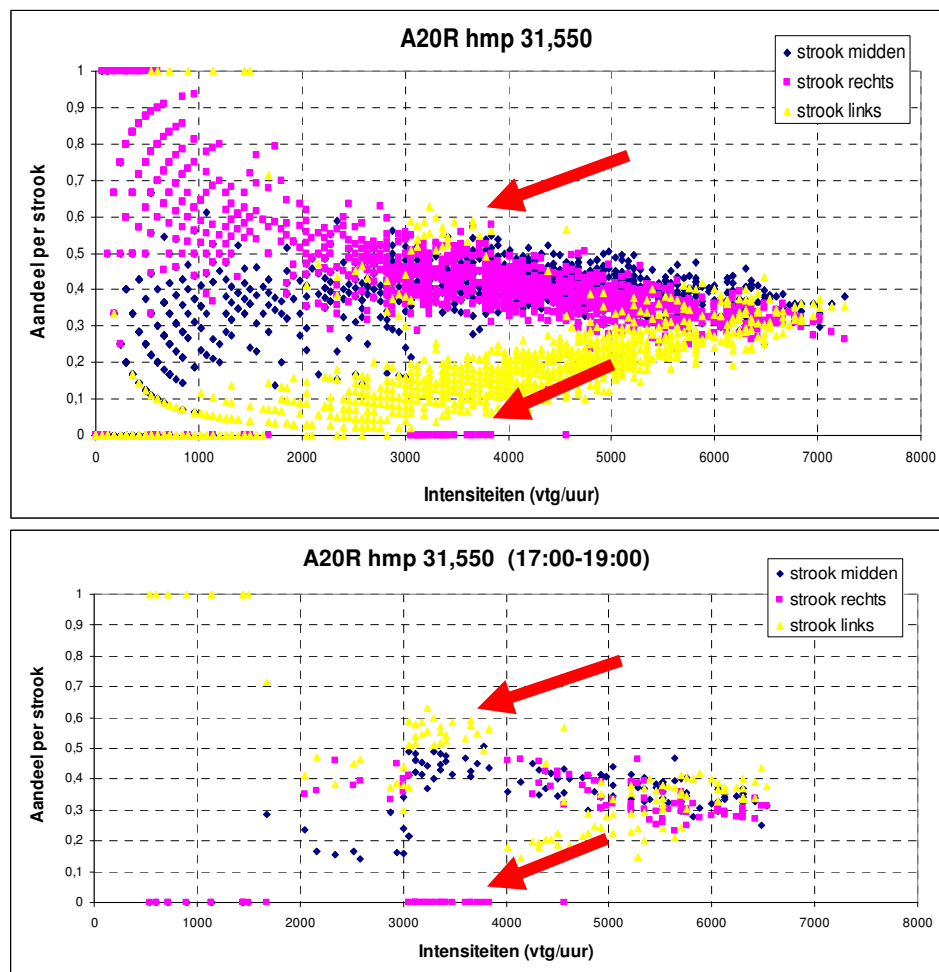
Het laatste onderzochte incident is een locatie waar de rijbaan drie rijstroken heeft. Ook op deze locatie is in onderstaand figuur een duidelijk patroon zichtbaar in de verdeling van het verkeer over de rijstroken.

- Bij lage rijbaanintensiteiten (tot 1800 vtg/uur) maakt het verkeer vrijwel alleen gebruik van de rechter en midden rijstrook
- Als de rijbaanintensiteiten toenemen, neemt het aandeel op de linker rijstrook langzaam toe. Ongeveer evenveel verkeer maakt gebruik van de rechter en midden rijstrook.
- Bij zeer hoge rijbaanintensiteiten (boven 6000 vtg/uur) verdeelt het verkeer zich gelijkmatiger over de drie rijstroken.

Bij de rode pijlen is een duidelijke afwijking zichtbaar in de verdeling. Er bevindt zich geen verkeer op de rechter rijstrook, terwijl het aandeel op de linker rijstrook fors groeit. In de onderste figuur is te zien dat deze afwijking zich heeft voorgedaan in de periode van het incident.

Figuur 5.3:

Aandeel verkeer per rijstrook per intensiteit klasse: bovenste figuur voor de gedurende de gehele dag en onderste figuur in de periode van het incident



5.4 Rijstrookverdelingen als referentie voor een incident

Uit de analyse van de incidenten uit de vorige paragraaf kan de conclusie worden getrokken dat er bij incidenten een afwijkend patroon zichtbaar is in de rijstrookverdeling. De mogelijkheid voor het detecteren van incidenten aan de hand van de rijstrookverdeling zal dus in sommige gevallen mogelijk zijn. De vraag hierbij is echter of de afwijking in de periode van een incident

uitzonderlijk genoeg is om incidenten te kunnen detecteren. Als afwijkingen die zich voordoen tijdens incidenten ook te constateren zijn in incidentvrije situaties zal het aantal valsmeldingen namelijk zeer hoog kunnen zijn. Om hier inzicht in te krijgen wordt in deze paragraaf de rijstrookverdelingen van vergeleken met rijstrookverdelingen van incidentvrije dagen.

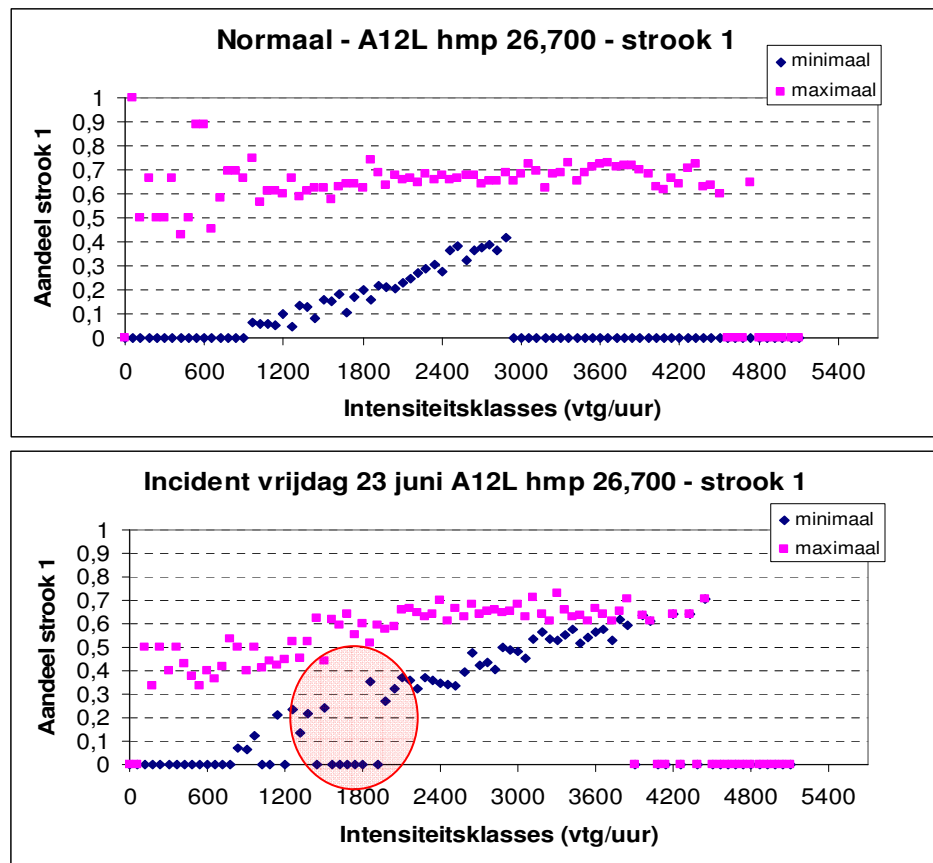
Om deze analyse uit te voeren worden gedurende de dag dat het incident heeft plaatsgevonden de maximale en minimale waarden voor het aandeel verkeer voor de betreffende strook bepaald voor elke intensiteit (dit zijn veelvoud van 60 aangezien per minuut de intensiteit wordt gemeten en we hier met vtg/uur rekenen). Dit wordt ook gedaan voor een aantal incidentvrije dagen. Deze gegevens worden vervolgens met elkaar vergeleken om te bepalen hoe uitzonderlijk de verdelingen tijdens incidenten zijn.

Incident 1: A12L hmp 26,700

In de grafieken op de volgende pagina zijn de resultaten te zien voor het eerste incident. De bovenste figuur geeft voor alle intensiteiten het minimale en maximale aandeel verkeer op de betreffende rijstrook in het geval van een normale situatie waarop geen incident heeft plaatsgevonden. De waarden voor de normale situatie zijn bepaald aan de hand van de meetwaarden van zes achtereenvolgende incidentvrije dagen. De onderste figuur geeft voor dezelfde locatie de minimale en maximale gemeten waarden weer voor de dag waarop het incident heeft plaatsgevonden. In de grafieken van de dag van het incident is een duidelijke afwijking zichtbaar (zie ronde markeringen) voor zowel rijstrook 1 als rijstrook 2.

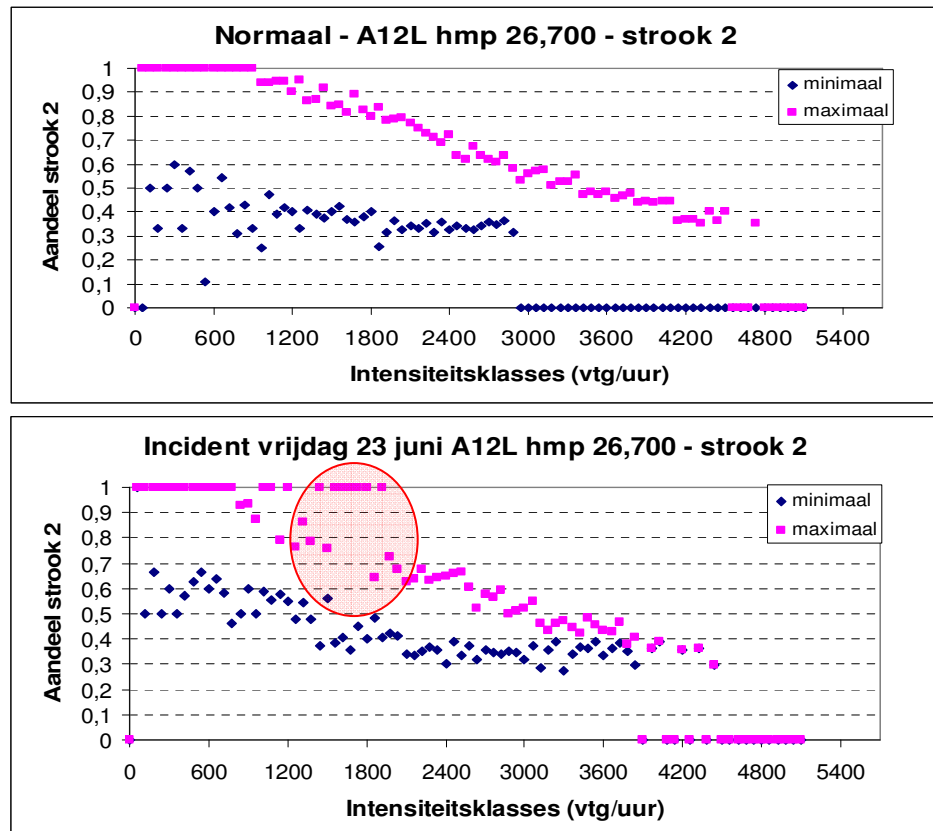
Figuur 5.4:

Grafiek van alle minimale en maximaal gemeten rijstrook aandelen per intensiteitsklasse voor strook 1: bovenste grafiek voor een normale dag en onderste grafiek voor een de dag van het incident.



Figuur 5.5:

Grafiek van alle minimale en maximaal gemeten rijstrook aandelen per intensiteitsklasse voor strook 2: bovenste grafiek voor een normale dag en onderste grafiek voor een de dag van het incident.



Uit bovenste grafieken kan men het volgende opmerken:

- Bij rijbaanintensiteiten lager dan 1000 vtg/uur is er geen sprake van een uitzonderlijke situatie tijdens het incident. Op de linker rijstrook (rijstrook 1) is het 'normaal' dat er geen verkeer van deze rijstrook gebruikt maakt. De maximale rijstrookaandelen liggen verspreid tussen 1,0 en 0,4. Uit deze grote spreiding in maximale waarden kan geconcludeerd worden dat het niet uitzonderlijk is dat bij deze rijbaanintensiteiten vrijwel alle verkeer gebruik maakt van de linker rijstrook. Voor de rechter rijstrook (rijstrook 2) worden in een normale situatie vaak waarden van 1,0 gemeten en liggen de minimale waarden zeer verspreid.
- Voor intensiteiten tussen de 1000 vtg/uur en 2800 vtg/uur is voor beide rijstroken en zowel de minimale als maximale waarden duidelijke grenswaarden vast te stellen waar de rijstrookverdeling tussen moet liggen. In het geval van een incident wijkt de rijstrookverdeling duidelijk af van deze minimale en maximale grenswaardes.
- Bij hoge rijbaanintensiteiten (hoger dan 2800 vtg/uur) liggen de minimale waarden op de linker rijstrook (strook 1) op nul en de maximale waarden rond 0,7. Op rijstrook 1 zullen aandelen hoger dan 0,7 uitzonderlijk genoeg zijn om incidenten te herkennen. Op rijstrook 2, de rechter rijstrook, liggen de minimale waarden op nul en de maximale waarden rond 0,6. Op deze rijstrook zijn dus rijstrookaandelen boven 0,6 uitzonderlijk genoeg.

Bij dit incident zal detectie aan de hand van rijstrookverdeling goede resultaten opleveren. De gemeten rijstrookverdelingen op de dag van het incident zijn uitzonderlijk genoeg en komen op incidentvrije dagen niet voor.

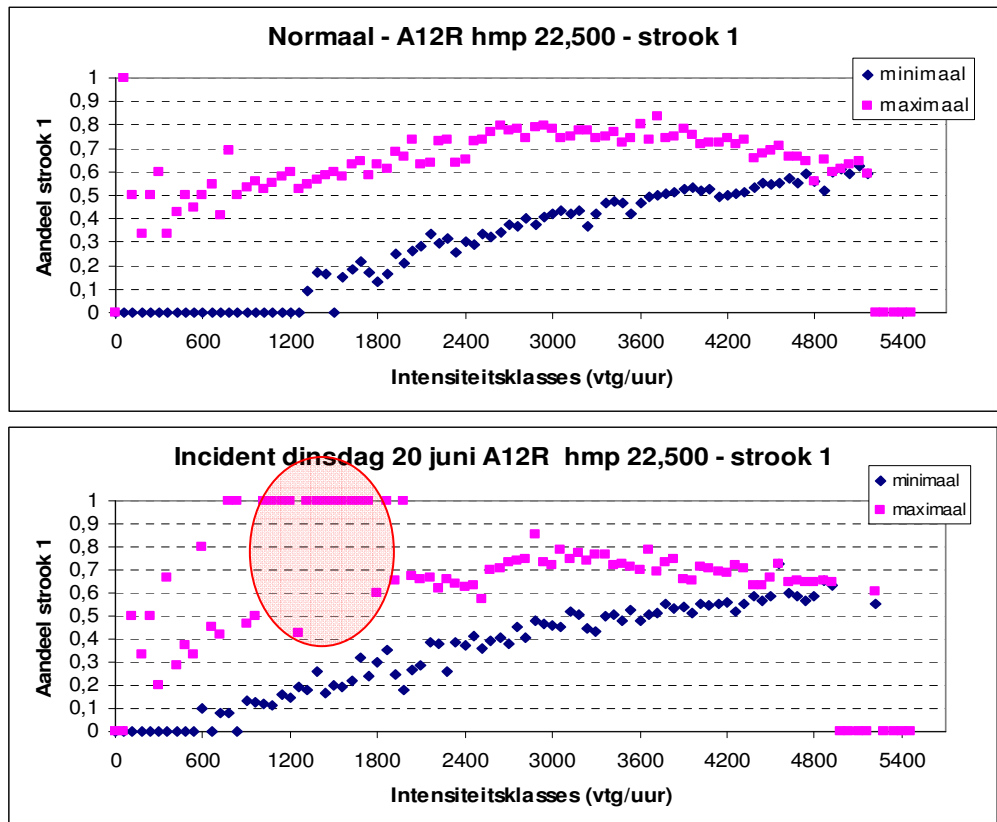
Incident 2: A12R hmp 20,500

In onderstaande figuur zijn ook voor deze locatie de minimale en maximale

rijstrookaandelen voor zowel een normale situatie (6 incidentvrije dagen) als voor een dag waar zich het incident heeft voorgedaan weergegeven. Ook hier is op de dag van het incident een duidelijke afwijking zichtbaar (zie gemarkeerde cirkel).

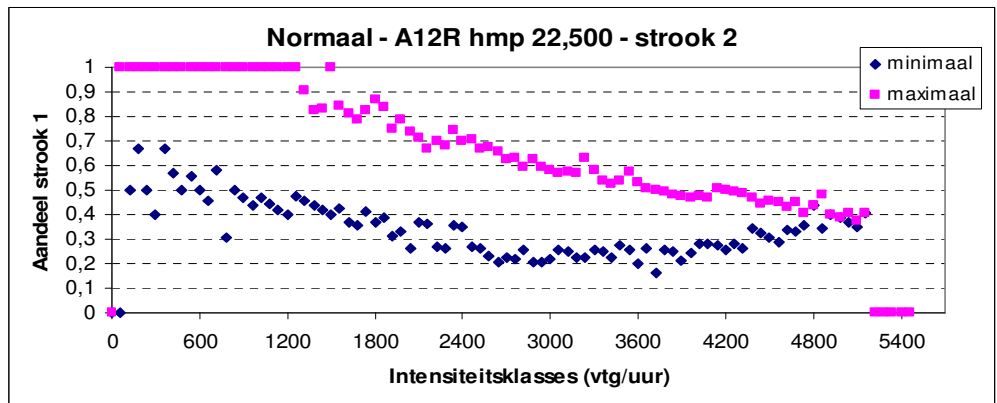
Figuur 5.6:

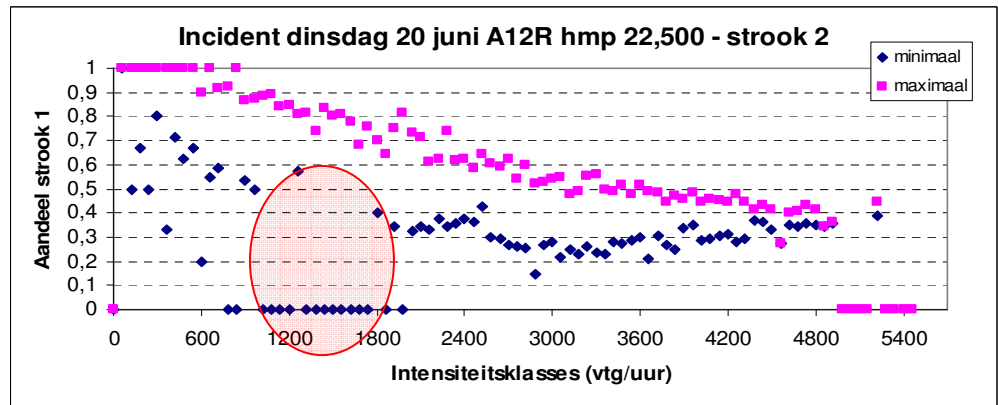
Grafiek van alle minimale en maximaal gemeten rijstrook aandelen per intensiteits-klasse voor strook 1: bovenste grafiek voor een normale dag en onderste grafiek voor een de dag van het incident.



Figuur 5.7:

Grafiek van alle minimale en maximaal gemeten rijstrook aandelen per intensiteits-klasse voor strook 2: bovenste grafiek voor een normale dag en onderste grafiek voor een de dag van het incident.





Uit bovenste grafieken kan men het volgende opmerken:

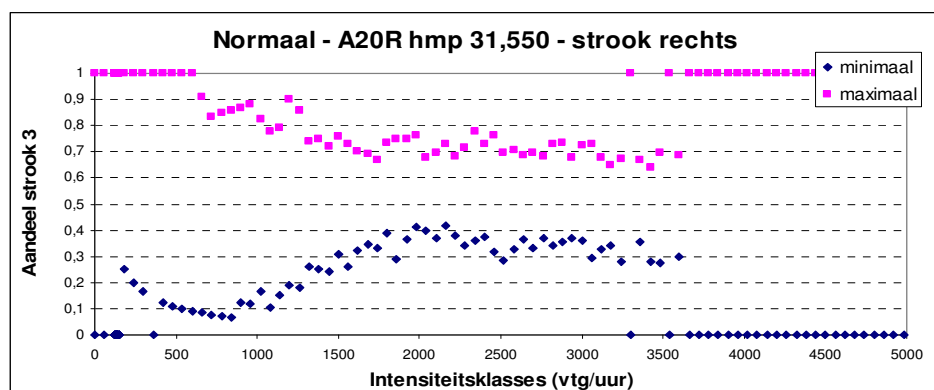
- Bij rijbaanincidenten lager dan 1300 voertuigen rijdt in incidentvrije dagen het verkeer vaak allemaal op de rechter rijstrook (rijstrook 2). Bij deze intensiteiten zijn waarden van 1,0 op rijstrook 2 en waarden van 0,0 op rijstrook 1 dan ook niet uitzonderlijk. Op rijstrook 1 liggen de maximale waarden bij deze intensiteiten verspreid tussen de 0,3 en 0,7. Waardes hoger dan 0,7 zouden dus als uitzonderlijk kunnen worden beschouwd. Op de rechter rijstrook liggen de minimale waarden verspreid tussen 0,7 en 0,3. Op deze rijstrook zullen waarden lager dan 0,3 dan als uitzonderlijk kunnen worden beschouwd.
- Voor rijbaanintensiteiten tussen de 1200 vtg/uur en 5000 vtg/uur is voor de incidentvrije situatie voor zowel de minimale als maximale waarden duidelijke grenswaarden te stellen. Er kan dan vanuit worden gedaan dat als de gemeten waarden niet tussen de grenswaarden liggen er sprake is van een incident.
- Bij hoge rijbaanintensiteiten (boven 5000) liggen de minimale en maximale waarden tussen de 0,0 en 1,0.

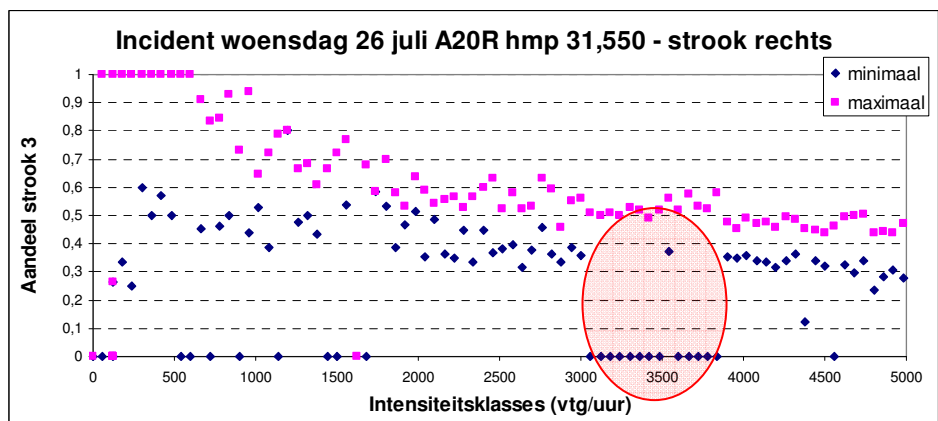
Incident 3: A20R hmp 31,550

De A20R bij hmp 31,550 bestaat de rijbaan uit drie rijstroken. In onderstaande figuren zijn voor alle drie de rijstroken de minimale maximale rijstrookverdelingen uit de meetwaarden weergegeven. Voor de normale situatie is uitgegaan van zes incidentvrije dagen. Op de rechter strook is een duidelijke afwijking te constateren op de dag van het incident (gemarkeerde cirkel). Op de andere rijstroken is geen duidelijke afwijking zichtbaar.

Figuur 5.8:

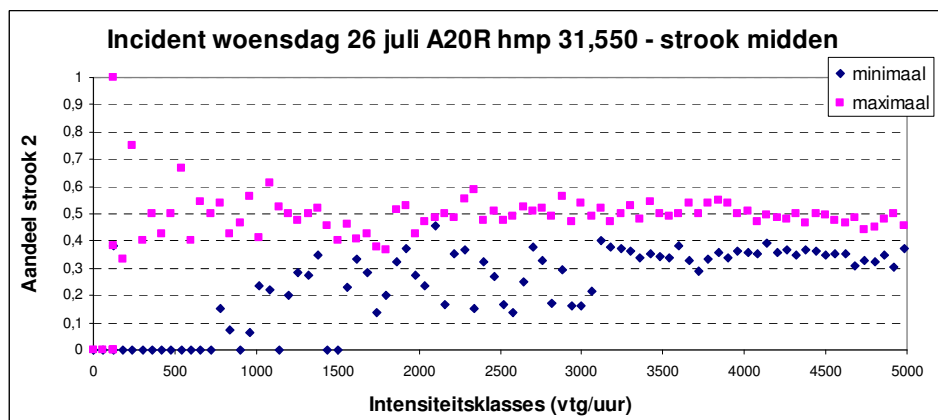
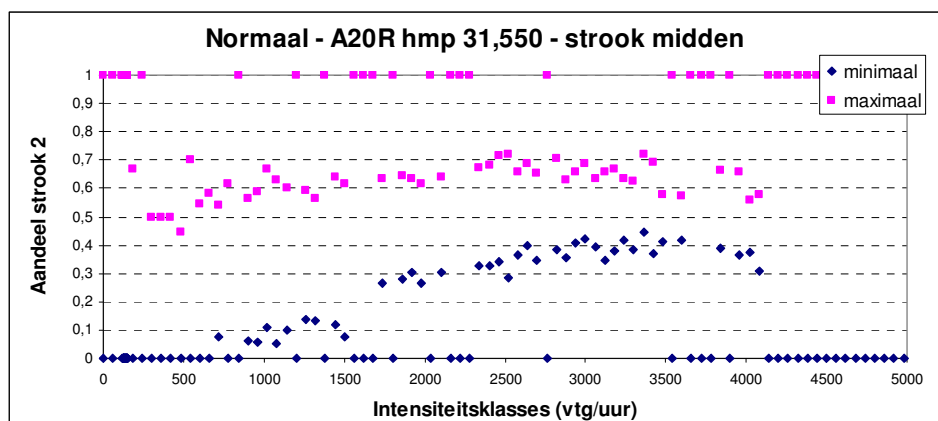
Grafiek van alle minimale en maximaal gemeten rijstrook aandelen per intensiteitsklasse voor strook rechts: bovenste grafiek voor een normale dag en onderste grafiek voor een de dag van het incident.





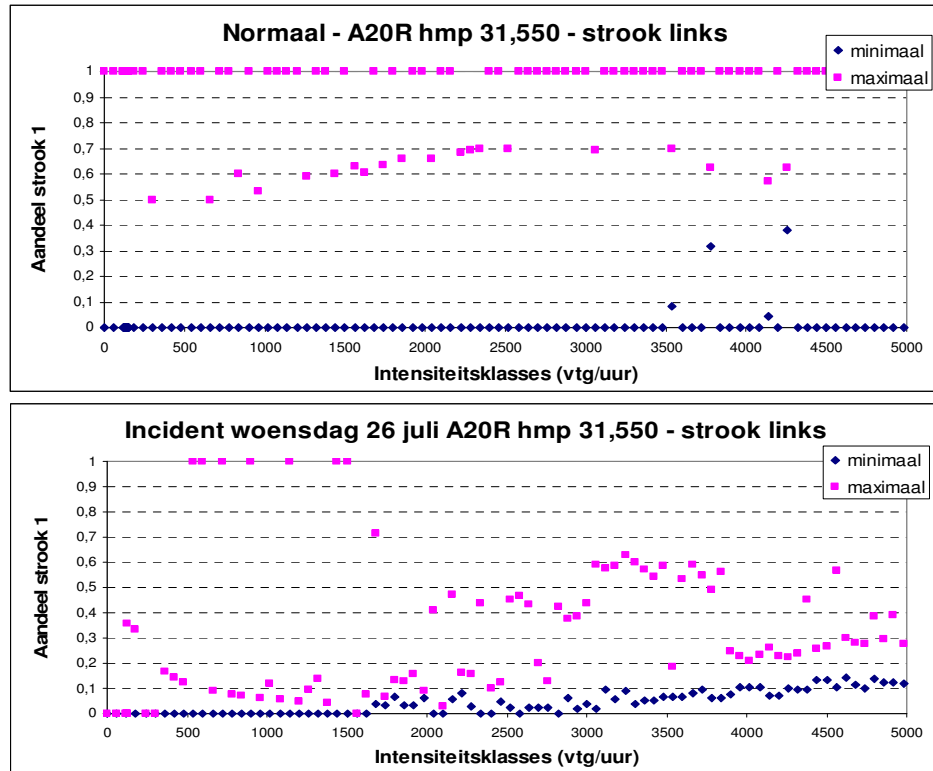
Figuur 5.9:

Grafiek van alle minimale en maximaal gemeten rijstrook aandelen per intensiteitsklasse voor strook midden: bovenste grafiek voor een normale dag en onderste grafiek voor een de dag van het incident.



Figuur 5.10:

Grafiek van alle minimale en maximaal gemeten rijstrook aandelen per intensiteitsklasse voor strook links: bovenste grafiek voor een normale dag en onderste grafiek voor een de dag van het incident.



Uit bovenste grafieken kan men het volgende opmerken:

- Alleen op de rechter rijstrook zijn duidelijke grenswaarden vast te stellen die zich voordoen op incidentvrije dagen. Op de linker rijstrook liggen voor vrijwel alle rijbaanintensiteiten de grenswaarden tussen 0,0 en 1,0. Op de middelste rijstrook zijn voor een aantal rijbaanintensiteiten wel duidelijke grenswaarden te stellen. Echter komen bij enkele intensiteiten minimale waarden van 0,0 voor en maximale waarden van 1,0. Bij deze rijstrook hoeven afwijkingen in de rijstrookverdeling dus niet zozeer te duiden op een incident.
- Op de rechter rijstrook zijn bij lage rijbaanintensiteiten (lager dan 900 vtg/uur) de maximale waarden 1,0 en de minimale waarden variëren tussen 0,3 en 0,05. Bij deze intensiteiten zullen geen duidelijkheden vastgesteld kunnen worden in het geval van een incident. Ook bij rijbaanintensiteiten boven de 3400 vtg/uur zijn geen bruikbare grenswaarden te bepalen. Bij intensiteiten tussen 900 vtg/uur en 3400 vtg/uur zijn wel duidelijke minimale en maximale waarden op te stellen.

Uit de analyse van de incidenten in de vorige paragrafen kan men concluderen dat:

- In het geval van een incident een duidelijke afwijking in de rijstrookverdeling te constateren is.
- Deze rijstrookverdeling is in het geval van een rijbaan met twee rijstroken voor beide rijstroken uitzonderlijk genoeg om een incident te detecteren. Aan de hand van incidentvrije dagen zijn grenswaarden op te stellen voor de minimale en maximale waarden per rijstrook.
- Indien de rijbaan uit meer dan twee rijstroken bestaat zijn voor de rechter rijbaan ook minimale en maximale grenswaarden te bepalen en is de rijstrookverdeling in het geval van een incident uitzonderlijk genoeg. Voor de middelste en linkse rijbanen geen goede

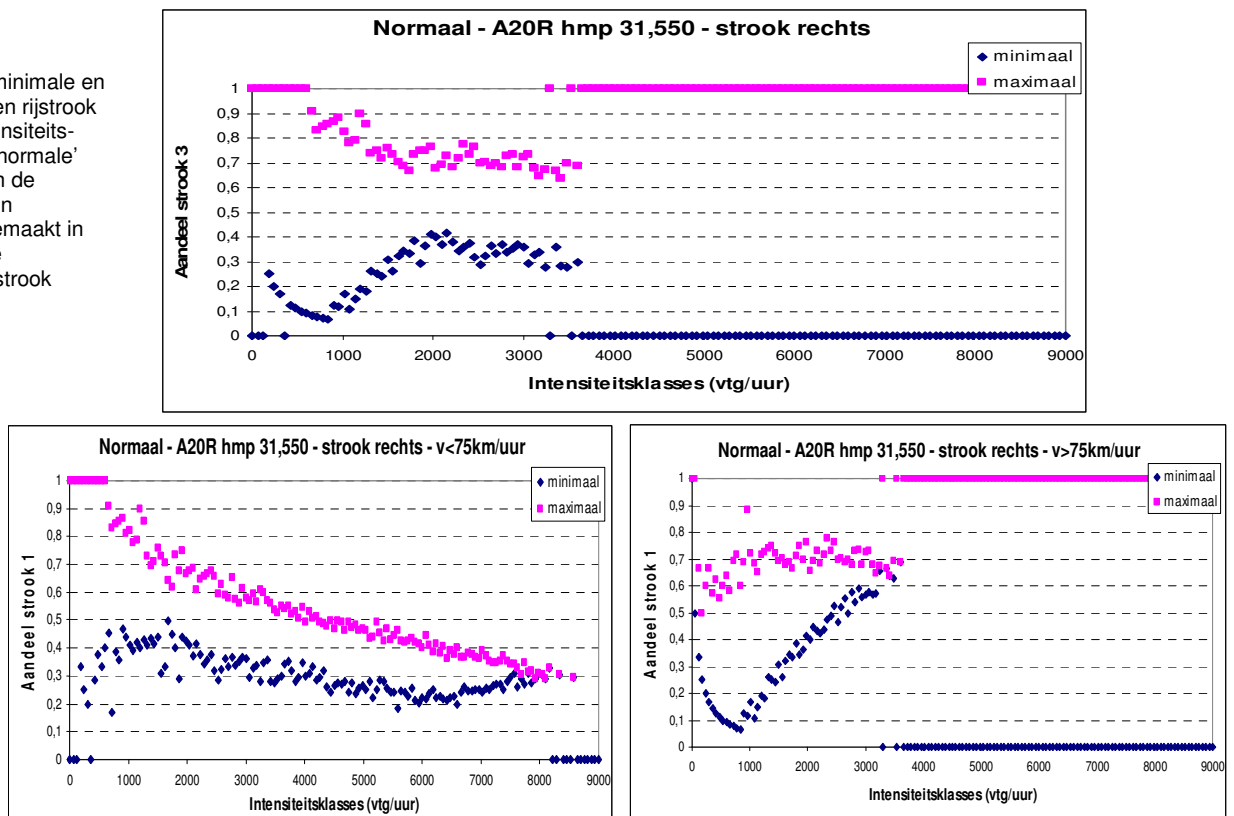
grenswaarden vast te stellen. Indien congestie en geen congestie wordt onderscheiden is hierin wellicht een beter beeld zichtbaar, dit wordt in de volgende paragraaf behandeld.

5.5 Onderscheid congestie en vrije afwikkeling

In onderstaande figuren (figuur 5.11 tot 5.13) is voor het incident op de A20 Rechts de grenswaarden per rijstrook weergegeven waarbij onderscheid is gemaakt tussen congestie en vrije afwikkeling. Hierbij is uitgegaan van congestie indien snelheden lager zijn dan 75 km/uur en van vrije afwikkeling indien de snelheid hoger of gelijk is aan 75 km/uur. De bovenste grafiek geeft alle minimale en maximale gemeten rijstrookaandelen aan en in de onderste twee grafieken is onderscheid gemaakt tussen vrije afwikkeling (snelheden hoger dan 75 km/uur) en congestie (snelheden lager dan 75 km/uur). Uit deze grafieken is te concluderen dat, voornamelijk voor de middelste en linker rijstrook, veel duidelijkere grenswaarden op te stellen zijn door verkeerssituaties waarin zich congestie voordoet en verkeerssituaties met vrije afwikkeling apart te beschouwen. Er zullen dus voor zowel grenswaarden moeten worden vastgesteld voor vrije afwikkeling als voor congestie.

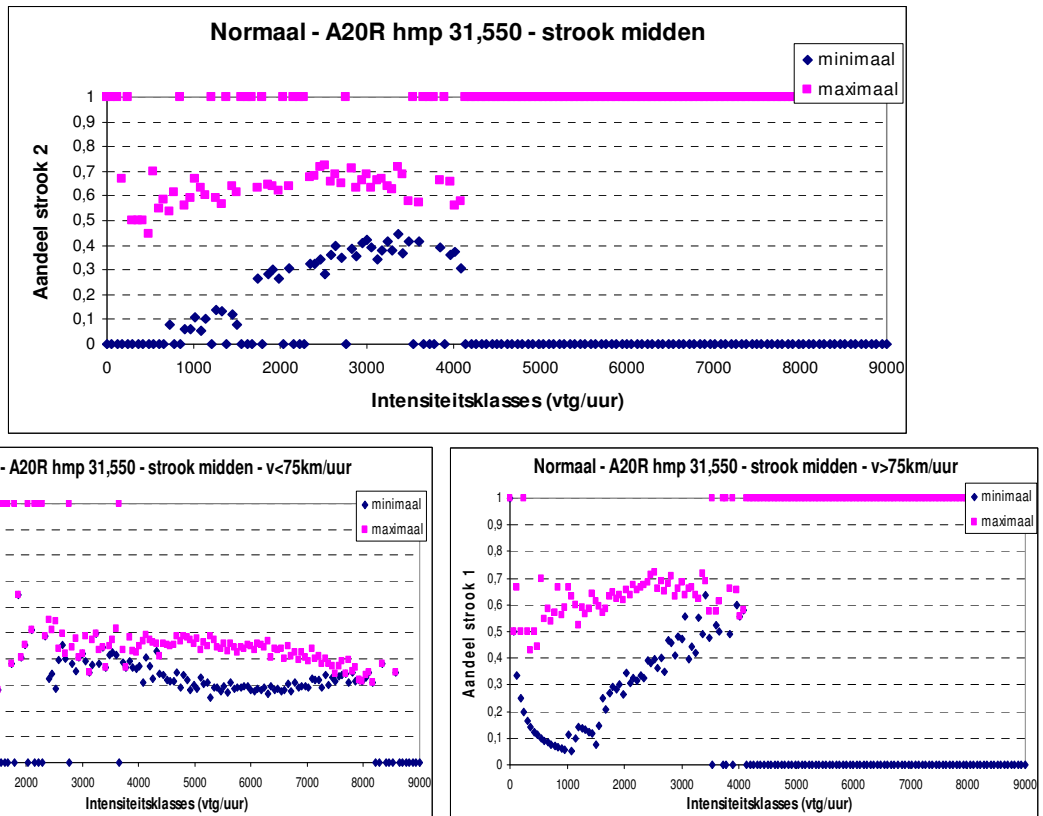
Figuur 5.11:

Grafiek van alle minimale en maximaal gemeten rijstrook aandelen per intensiteitsklasse voor een 'normale' situatie, waarbij in de onderste grafieken onderscheid is gemaakt in congestie en vrije afwikkeling voor strook rechts



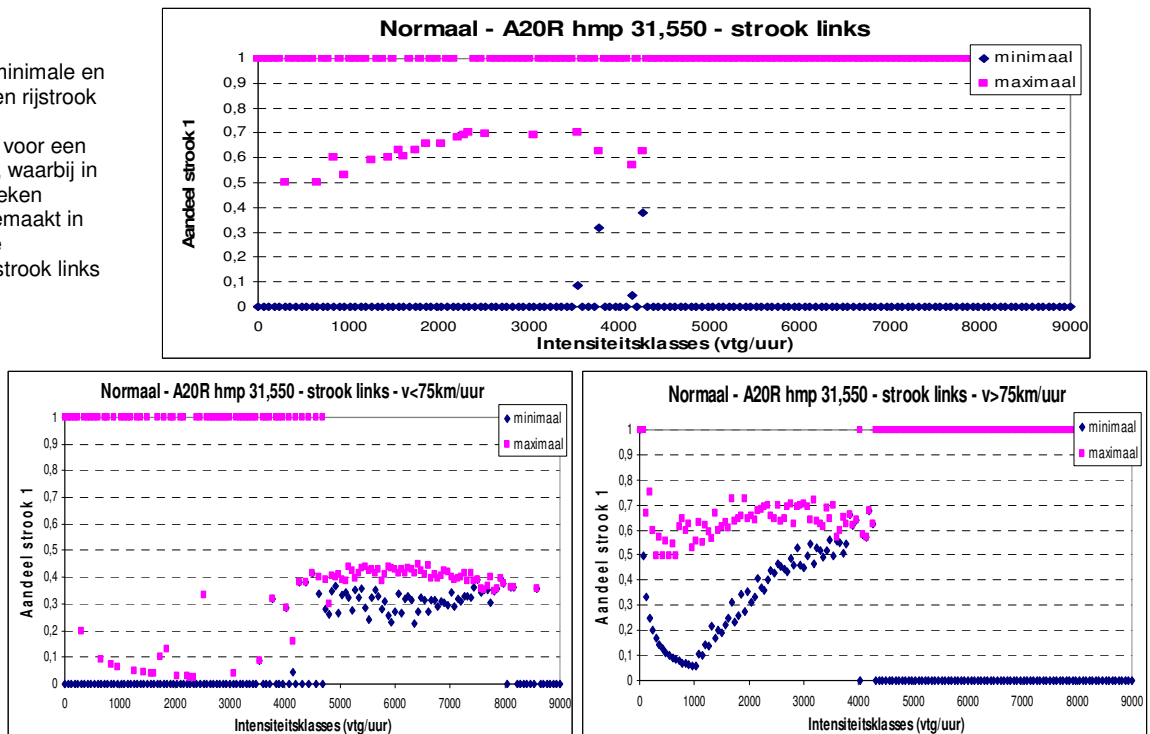
Figuur 5.12:

Grafiek van alle minimale en maximaal gemeten rijstrook aandelen per intensiteitsklasse voor een 'normale' situatie, waarbij in de onderste grafieken onderscheid is gemaakt in congestie en vrije afwikkeling voor strook midden



Figuur 5.13:

Grafiek van alle minimale en maximaal gemeten rijstrook aandelen per intensiteitsklasse voor een 'normale' situatie, waarbij in de onderste grafieken onderscheid is gemaakt in congestie en vrije afwikkeling voor strook links



6. Incidentdetectie methode op rijstrookniveau

In het vorige hoofdstuk is duidelijk naar voren gekomen dat tijdens incidenten de rijstrookverdeling afwijkend is. Ook blijkt dat deze afwijking in bepaalde intensiteitsklassen uitzonderlijk genoeg is om incidenten te kunnen herkennen. Verder is het verschil in rijstrookverdeling in geval van congestie en geen congestie groot. In dit hoofdstuk wordt, aan de hand van de conclusies die uit hoofdstuk 5 getrokken kunnen worden, een nieuwe incident detectie methode voorgesteld op rijstrookniveau. De opzet van deze methode is een eerste idee. Om de methode optimaal te kunnen laten functioneren, zal verder onderzoek moeten worden gedaan om onder andere de parameters nauwkeurig vast te stellen.

6.1 Uitgangspunten bij ontwikkelen detectiemethode op rijstrookniveau

In deze paragraaf worden kort de belangrijkste resultaten uit het vorige hoofdstuk weergegeven. Deze resultaten dienen als uitgangspunt voor het ontwikkelen van een nieuw algoritme op rijstrookniveau.

- In het geval van een incident een duidelijke afwijking in de rijstrookverdeling te constateren.
- Deze rijstrookverdeling is in het geval van een rijbaan met twee rijstroken voor beide rijstroken uitzonderlijk genoeg om een incident te detecteren. Aan de hand van incidentvrije dagen zijn grenswaarden op te stellen voor de minimale en maximale waarden per rijstrook. Deze grenswaarden zijn locatie en strook afhankelijk en moeten voor elke locatie en strook apart bepaald worden. Voor intensiteiten lager dan 500 vtg/uur per rijstrook is geen duidelijk patroon zichtbaar in de verdeling van het verkeer over de rijstroken. Het zelfde geldt voor rijbaanintensiteiten zeer hoog zijn.
- Vooral voor rijbanen met meer als twee rijstroken is het van belang om de grenswaarden in het geval van congestie en vrije afwikkeling apart op te stellen.

Gezien deze resultaten wordt er een idee opgezet voor een mogelijke manier om een detectiemethode te ontwikkelen op rijstrookniveau. In de volgende paragraaf wordt deze methode uitgelegd.

6.2 Beschrijving detectie algoritme op strookniveau

Het algoritme dat hier wordt voorgesteld maakt gebruik van gegevens over rijstrookgebruik. De gemeten rijstrookaandelen worden vergeleken met waarden die op een bepaalde locatie op 'normale' dagen voorkomen. Aangezien de verdeling van het verkeer over de rijstroken afhankelijk is van de rijbaanintensiteit op dat moment, worden de grenswaarden opgesteld afhankelijk van de rijbaanintensiteit. De grenswaarden worden bepaald aan de

hand van de minimale en maximale gemeten rijstrookaandelen op een aantal incidentvrije dagen. De eerste stap in dit algoritme is dan ook het vaststellen van de grenswaarden voor de rijstrookverdeling. Deze grenswaarden zijn locatieafhankelijk en worden per meetlocatie bepaald.

Voor het bepalen van deze grenswaarden wordt onderscheid gemaakt tussen congestie en een vrije verkeersafwikkeling. Als de grenswaarden bepaald zijn wordt voor elke minuut en locatie de rijstrookverdeling vergeleken met de grenswaarden van die rijbaanintensiteit. Indien de rijstrookverdeling afwijkt van de grenswaarden wordt een melding van een incident gegeven.

Als intensiteiten lager zijn dan ongeveer 500 vtg/uur per strook wordt deze uitgesloten. Bij lage intensiteiten is de minuutmeting gebaseerd op een laag aantal voertuigen (minder dan 10) en zijn er grote afwijkingen in het rijstrookgebruik niet uitzonderlijk. Bij een intensiteit van bijvoorbeeld 600 vtg/uur zijn dus in totaal 6 voertuigen gedetecteerd. Indien 5 van deze voertuigen op een bepaald moment een voertuig gaan inhalen die op de rechter rijstrook rijdt, is het aandeel verkeer op de linker rijstrook dus 83%. Dit is geen uitzonderlijke situatie, maar zal anders wel als zodanig gedetecteerd worden. Ook als de rijbaanintensiteit zeer hoog is wordt deze uitgesloten in dit algoritme.

Beschrijving algoritme:

Onderstaand wordt het algoritme stapsgewijs weergegeven.

Stap 1:

De eerste stap in dit algoritme is het vaststellen van de grenswaarden voor de minimale en maximale strookaandelen in een 'normale' situatie. Deze waarden worden bepaald door voor een periode van een aantal incidentvrije dagen de minimale en maximale gemeten rijstrookaandeel per intensiteit te bepalen voor elke strook. Hierbij worden aparte grenswaarden vastgesteld voor congestie en vrije afwikkeling. Deze grenswaarden kunnen online vastgesteld worden aan de hand van incidentvrije dagen van de afgelopen periode.

Stap 2

In de tweede stap van het algoritme wordt per rijstrook het aandeel verkeer berekend en dit wordt getoetst aan de grenswaarden. De verdeling van het verkeer over de rijstroken zou in een normale situatie immers tussen de minimale en maximale grenswaarden moeten liggen. Eerst wordt uit de rijstrookgegevens de rijbaanintensiteit bepaald. De rijbaanintensiteit is een sommatie van de gemeten intensiteiten van alle rijstroken op die locatie:

$$q(t, x) = \sum_r q_r(t, x) \quad [6.1]$$

waarin: $q(t, x)$ = rijbaanintensiteit op tijdstip t en locatie x
 $q_r(t, x)$ = rijstrookintensiteit op tijdstip t en locatie x

Aan de hand van de rijbaanintensiteit en rijstrookintensiteit kan per strook het aandeel verkeer worden berekend:

$$A_r(t, x) = \frac{q_r(t, x)}{q(t, x)} \quad [6.2]$$

waarin: $A_r(t, x)$ = aandeel verkeer op strook r op tijdstip t en locatie x
 $q(t, x)$ = rijbaanintensiteit op tijdstip t en locatie x
 $q_r(t, x)$ = rijstrookintensiteit op tijdstip t en locatie x

Stap 3:

In de derde stap wordt bepaald of de strook en locatie in de intensiteitsklassen

ligt waarvoor dit algoritme werkt. Hiervoor moeten onderstaande voorwaarden gelden:

- de rijbaanintensiteit moet kleiner zijn dan een parameterwaarde voor de minimale rijbaanintensiteit en maximale intensiteit:

$$r * par_min_q < q(t, x) < r * par_max_q \quad [6.3]$$

waarin: r = aandeel rijstroken
 $q(t, x)$ = rijbaanintensiteit op tijdstip t en locatie x
 par_min_q = parameter minimale rijbaanintensiteit
 par_max_q = parameter maximale rijbaanintensiteit

Als aan bovenstaande voorwaarden wordt voldaan wordt bepaald of er op dat moment sprake is van vrije afwikkeling of congestie. Hierbij wordt in dit onderzoek ervan uitgegaan dat bij snelheden lager dan 75 km/uur sprake is van congestie en bij snelheden groter dan 75 km/uur sprake is van vrije afwikkeling. Als de verkeerstoestand is bepaald wordt voor elke strook gecontroleerd of de gemeten rijstrookaandeel tussen de grenswaarden ligt. Hierbij worden de parameters k_1 en k_2 toegevoegd om te veel vals meldingen uit te sluiten.

$$k_1 * min_grenswaarde < q_r(t, x) < k_2 * max_grenswaarde \quad [6.4]$$

waarin: $q_r(t, x)$ = rijbaanintensiteit op tijdstip t en locatie x
 $min_grenswaarde$ = parameter minimale rijbaanintensiteit
 k_1 = afwijking naar beneden
 $max_grenswaarde$ = parameter maximale rijbaanintensiteit
 k_2 = afwijking naar boven

Als de waarde bovenstaande voorwaarde geldt dan wordt er een melding van een incident gegeven. Voor dit algoritme is dus geen vooraf ingestelde randvoorwaarden aanwezig. De grenswaarden voor de rijstrookverdelingen zijn locatie afhankelijk en moeten voor elke locatie en strook apart bepaald worden.

7. Testen en vergelijken algoritmen

In dit hoofdstuk worden de in hoofdstuk 4 besproken algoritmen en het nieuw ontwikkelde algoritme op strookniveau geïmplementeerd en op kleine schaal getest. De test wordt uitgevoerd op de A12 links voor de maanden juni en juli. In de eerste paragraaf worden de indicatoren geïntroduceerd die zijn gebruikt voor het vergelijken van de methodes. In paragraaf 7.2 wordt ingegaan op de incidentmeldingen die door VCNL zijn gerapporteerd. In de daaropvolgende paragraaf wordt de implementatie van de algoritmes besproken. In de paragrafen 7.4 en 7.5 worden respectievelijk de resultaten en de conclusies en vergelijkingen van de test weergegeven en besproken.

7.1 Indicatoren voor vergelijking

Voor het valideren van de detectiemethoden wordt gebruik gemaakt van een aantal indicatoren die de prestatie van de detectie weergeven. Deze drie indicatoren zijn:

- *Detectietijd*: de gerapporteerde detectietijd is gelijk aan het verschil tussen de starttijd van het incident zoals deze staat aangegeven in de lijsten van VCNL en de tijd waarop de detectiemethode voor het eerst een melding geeft. Een negatieve detectietijd houdt dus in dat de methode het incident eerder meldt dan de melding bij de verkeerscentrales is binnengekomen.
- *Volledigheid*: de gerapporteerde volledigheid is het percentage incidenten dat door de methode juist is gemeld (dus zonder valsmeldingen) in verhouding tot het aantal incident dat VCNL in die periode meldt.
- *Aantal valsmeldingen*: de gerapporteerde aantal foutmeldingen is het aantal valsmeldingen dat wordt gegeven.

Naast deze kwantitatieve zijn ook aspecten als de complexiteit van het inregelen, zoals het instellen van de parameters en de inzetbaarheid van de methode belangrijke indicatoren. Bij de inzetbaarheid van de methoden gaat om de gevoeligheid van bijvoorbeeld op- en afritten of het aantal rijstroken.

7.2 De incidenten

In totaal zijn er in de maanden juni en juli 25 incidenten gerapporteerd door VCNL. Voor vijf van deze incidenten was geen verkeersdata beschikbaar. Voor de andere incidenten is nagegaan of deze ook zichtbaar waren in de intensiteitsgegevens en snelheidsgegevens. Hierbij is voornamelijk gekeken of er afsluitingen van rijstroken te vinden waren waar de intensiteiten tijdelijk nul waren. In onderstaande tabel zijn deze bevindingen weergegeven.

Tabel 7.1 Resultaten detectiealgoritmes

datum	Weg	hmp	Begintijd	beginmi nuut	omschrij ving	Opmerkingen uit verkeersgegevens
5-6	A12L	16,60	11:43	704	Verkeers ongeval	Goed zichtbaar (afsluiting strook 1)
5-6	A12L	13,10	13:34	815	Verkeers ongeval	Goed zichtbaar (afsluiting strook 1)
6-6	A12L	13,10	17:02	1023	Kijkfile	Niks duidelijk zichtbaar
6-6	A12L	27,00	18:49	1130	Verkeers ongeval	Redelijk zichtbaar (strook 4)
12-6	A12L	25,40	16:43	1004	Gestrand voertuig	Goed zichtbaar (strook 2)
14-6	A12L	25,40	16:39	1000	Verkeers ongeval	Goed zichtbaar (strook 3 en deel 4)
14-6	A12L	29,60	16:44	1005	Overige oorzaken	Niks duidelijk zichtbaar, waarschijnlijk zelfde incident als vorige
15-6	A12L	8,50	8:59	540	Verkeers ongeval	Niks duidelijk zichtbaar
20-6	A12L	4,40	7:42	463	Verkeers ongeval	Niks zichtbaar (wel een afsluiting van strook 3 op tijdstip 553)
20-6	A12L	29,60	8:27	508	Verkeers ongeval	Zeer weinig zichtbaar
22-6	A12L	37,60	8:21	502	Verkeers ongeval	Goed zichtbaar (afsluiting strook 1)
23-6	A12L	26,70	12:33	754	Verkeers ongeval	Goed zichtbaar (afsluiting strook 1, wel vroeger op 712)
27-6	A12L	6,60	7:52	473	Kijkfile	Niks duidelijk zichtbaar,
27-6	A12L	37,60	8:07	488	Gestrand voertuig	Goed zichtbaar (afsl. strook 3), ook op 391 een afsl. van een strook zichtbaar
27-6	A12L	28,00	8:45	526	Verkeers ongeval	Redelijk zichtbaar (afsluiting strook 4)
30-6	A12L	13,10	8:48	529	Verkeers ongeval	Kort zichtbaar
30-6	A12L	19,50	9:20	561	Verkeers ongeval	Redelijk zichtbaar
2-7	A12L	22,30	10:29	630	Verkeers ongeval	Goed zichtbaar (afsluiting strook)
6-7	A12L	27,00	17:24	1045	Verkeers ongeval	Geen data
6-7	A12L	29,60	18:07	1088	Verkeers ongeval	Geen data
7-7	A12L	27,00	17:58	1079	Verkeers ongeval	Geen data
7-7	A12L	37,60	19:22	1163	Verkeers ongeval	Geen data
8-7	A12L	8,50	13:45	826	verkeerso ongeval	Geen data
29-7	A12L	29,60	14:00	841	verkeerso ongeval	Goed zichtbaar (afsluiting strook 3, wel vroeger op 828)
29-7	A12L	29,60	14:35	876	verkeerso ongeval	Niks zichtbaar, waarschijnlijk zelfde incident als vorige

Uit de bevindingen die in tabel 7.1 staan weergegeven, kan opgemerkt worden veel van de meldingen in de verkeersdata terug te vinden is. Beide ‘incidenten’ met als omschrijving “kijkfile” kunnen niet duidelijk herkend worden. Daarnaast is in twee gevallen (op 14 juni en 29 juli) op bijna dezelfde tijd en locatie een melding gegeven. Hoogstwaarschijnlijk is in deze gevallen twee maal een incident melding gerapporteerd van het zelfde incident. Deze beide incidenten, de twee kijkfiles en de incidenten waarvoor geen data beschikbaar was, worden niet meegenomen in de test. In totaal zijn er dus 16 incidentmeldingen.

7.3 Implementatie van de detectiealgoritmen

Het programma MoniGraph wordt gebruikt om de algoritmes te kunnen testen en vergelijken. In deze paragraaf wordt voor elke algoritme ingegaan op de implementatie van de algoritmes. Voor de drie al bestaande algoritmes, de Blokkadedetector, het Presikhaaf algoritme en het McMaster algoritme zijn voor het kalibreren tests uitgevoerd om optimale parameterinstellingen te bepalen. Voor het Strook algoritme worden de parameters bepaald aan de hand van de bevindingen uit de analyse van de incidenten uit hoofdstuk 5.

7.3.1. Implementatie Blokkadedetector

De blokkadedetector is voor eerder onderzoek al geïmplementeerd binnen Matlab voor gebruik in MoniGraph (Knibbe, AVV). Hiervan is voor dit onderzoek dan ook gebruik gemaakt. In bijlage C is de Matlab code van dit algoritme toegevoegd.

Voor het verifiëren van de code is het programma met behulp van breakpoints doorgelopen om de juistheid van de code te bepalen.

Voor het kalibreren, het bepalen van de parameters voor de blokkadedetector is voor een aantal mogelijke parameters een test uitgevoerd. Voor de periode van 1 juni tot 15 juni is voor de parametersamenstellingen die in tabel 7.2 staan weergegeven voor de A12 links de detectietijd, de betrouwbaarheid en het aantal gemiste meldingen bepaald. De parameters in paratest 1 zijn de parameters zoals deze zijn beschreven in het rapport van Transpute. De parameters uit de andere test zijn bepaald aan de hand van testresultaten. Indien bleek dat bepaalde parameters te gevoelig of te ongevoelig stonden ingesteld, zijn deze aangepast. De resultaten van alle parameter test worden hieronder beschreven. Hierbij is onderscheidt gemaakt tussen het detecteren in een stap (ahv H_q_klein), detectie in twee stappen (ahv G_q_klein) en het detecteren in drie stappen (ahv H_q_groot, G_q_groot, F_q_melden, F_v_melden en V_checken).

Tabel 7.2 Gebruikte parameters voor de kalibratie van de Blokkadedetector

	paratest 1	paratest 2	paratest 3	paratest 4
H_q_klein	0,20	0,00	0,10	0,05
G_q_klein	0,20	0,00	0,10	0,05
H_q_groot	0,80	0,90	0,90	0,90
G_q_groot	0,80	0,90	0,90	0,90
F_q_melden	0,50	0,80	0,50	0,70
F_v_melden	0,50	0,80	0,50	0,50
V_checken	50,00	50,00	50,00	50,00

Detectie in check 1: H_q_klein

In de eerste stap van de blokkadedetector kan met behulp van de parameter H_q_klein worden vastgesteld of het met voldoende zekerheid een incident in deze stap kan worden vastgesteld. In onderstaande tabel zijn deze resultaten weergegeven. Uit deze resultaten kan geconcludeerd worden dat het detecteren van incidenten in de eerste check aan de hand van H_q_klein geen goede resultaten oplevert. Er wordt immers maximaal een zeer laag percentage incidenten gedetecteerd (33%), met een detectietijd van 9 minuten en het aantal valsmeldingen is zeer hoog (129).

Tabel 7.3 Resultaten parameter tests Blokkadedetector in een stap a.h.v. H_q_klein

		Paratest 1 0,20		paratest 2 0,00		paratest 3 0,10		paratest 4 0,05	
detectie in stap 1	gedetecteerde meldingen (totaal 6 incidenten)	aantal	%	aantal	%	aantal	%	aantal	%
		2	33	0	0	0	0	0	0
	Betrouwbaarheid (aantal valse meldingen)	129		0		19		0	
	detectietijd (aantal minuten na melding)	9		0		0		0	

Detectie in check 2: G_q_klein

Het vaststellen van een incident in twee stappen kan aan de hand van de parameter G_q_klein. In onderstaande tabel zijn de resultaten weergegeven. Ook uit deze resultaten kan men concluderen dat het detecteren in deze stap bij een laag detectiepercentage van 33% van de incidenten nog altijd 8 valsmeldingen geeft en een detectietijd van 6 minuten. Gezien deze resultaten wordt besloten dat voor test van de blokkadedetector in alle gevallen drie stappen worden uitgevoerd voordat een incidentmelding gegeven wordt.

Tabel 7.4 Resultaten parameter tests Blokkadedetector in twee stappen a.h.v. G_q_klein

		paratest 1 0,20		paratest 2 0,00		paratest 3 0,10		paratest 4 0,05	
detectie in stap 2	gedetecteerde meldingen (totaal 6 incidenten)	Aantal	%	aantal	%	aantal	%	aantal	%
		2	33	0	0	2	33	0	0
	betrouwbaarheid (aantal valse meldingen)	980		0		8		4	
	detectietijd (aantal minuten na melding)	5		0		6		0	

Detectie in check 3

In tabel 7.5 zijn de resultaten van de parameters die getest zijn voor de blokkadedetector weergegeven indien alle drie de stappen worden uitgevoerd. De parameters uit paratest 4 geven de beste resultaten en deze zullen dan ook worden ingesteld voor de test. In vergelijking met paratest 3 word een veel hoger percentage incidenten gedetecteerd (83 % tegenover 50%), terwijl 3 valsmeldingen meer in 15 dagen acceptabel is.

Tabel 7.5 Resultaten parameter tests Blokkadedetector in drie stappen ahv H_q_groot, G_q_groot, F_q_melden, F_v_melden en V_checken

		paratest 1		paratest 2		paratest 3		paratest 4	
		Aantal	%	aantal	%	aantal	%	aantal	%
9detectie in stap 3	gedetecteerde meldingen (totaal 6 incidenten)	1	17	6	100	3	50	5	83
	betrouwbaarheid (aantal valse meldingen)	2		838		6		9	
	detectietijd (aantal minuten na melding)	-22		-8,3		-11,3		-8,2	

7.3.2. Implementatie Presikhaaf algoritme

De eerste stap is het implementeren van dit algoritme binnen MoniGraph in Matlab om te kunnen testen. In bijlage C is de code van dit algoritme toegevoegd. Het verifiëren van de code is op de dezelfde wijze uitgevoerd als bij de Blokkadedetector. Met behulp van breakpoints is de code doorgelopen om de juistheid te bepalen.

Voor het kalibreren, het bepalen van de parameters voor dit algoritme, zijn een aantal parameters getest voor dezelfde periode als de blokkadedetector (1 juni tot 15 juni op A12 links). In tabel 7.6 zijn de parametercombinaties die getest zijn weergegeven.

Tabel 7.6 Gebruikte parameters voor de kalibratie van het Presikhaaf algoritme

	Paratest 1	Paratest 2	Paratest 3
stap 1.1	0,60	0,60	0,60
stap 1.2	75	75	75
stap 1.3	80	80	80
stap 2.1	0,80	0,80	0,80
stap 2.2	70	70	70
stap 2.3	0,80	0,95	0,95
stap 3.1	80,00	80,00	80,00
stap 3.2	1,50	1,50	0,50

In de eerste test (paratest 1) zijn de parameters gebruikt zoals deze door Pieter

van Dommele zijn voorgesteld. Indien uit de parameter test bleek dat parameters te gevoelig of te ongevoelig stonden ingesteld, zijn deze aangepast om te onderzoeken of daardoor betere resultaten te behalen zijn. Bij de resultaten is gekeken naar resultaten van detectie na 1 stap, na 2 stappen en na 3 stappen. In onderstaande tabellen zijn deze resultaten weergegeven. Bij detectie in 1 stap wordt alleen de controles uit stap 1 uitgevoerd, bij detectie in 2 stappen de controles tot stap 2.3 en bij detectie in stap 3 worden alle controles uitgevoerd.

Tabel 7.7 Resultaten parameter tests Presikhaaf algoritme bij detectie in de eerste stap

		para1		para2		para3	
detectie in stap 1	gedetecteerde meldingen (totaal 6 incidenten)	aantal	%	aantal	%	aantal	%
		2	33	4	67	4	67
	Betrouwbaarheid (aantal valse meldingen)	64		64		65	
	detectietijd (aantal minuten na melding)	-16		-15,5		-15,5	

Tabel 7.8 Resultaten parameter tests Presikhaaf algoritme bij detectie in twee stappen

		para1		para2		para3	
detectie in stap 2	Gedetecteerde meldingen (totaal 6 incidenten)	aantal	%	aantal	%	aantal	%
		2	33	3	50	3	50
	Betrouwbaarheid (aantal valse meldingen)	8		19		19	
	Detectietijd (aantal minuten na melding)	-7		-15,3		-15,3	

Tabel 7.9 Resultaten parameter tests Presikhaaf algoritme bij detectie in drie stappen

		para1		para2		para3	
detectie in stap 3	Gedetecteerde meldingen (totaal 6 incidenten)	Aantal	%	aantal	%	aantal	%
			2	33	3	50	3

Betrouwbaarheid (aantal valse meldingen)	8	17	1
Detectietijd (aantal minuten na melding)	-6	-14,3	-11,3

In tabel 4.6 staan de resultaten voor als alleen stap 1 wordt uitgevoerd. In het beste geval wordt 67% van het aantal incidenten gedetecteerd. Het aantal valsmeldingen is dan wel hoog (64 en 65). Het detecteren van incidenten door alleen stap 1 uit te voeren is dus niet voldoende. De resultaten in tabel 4.5 en tabel 4.6 geven respectievelijk de resultaten weer bij een detectie na twee stappen en na drie stappen. Het aantal valsmeldingen is na twee stappen nog steeds vrij hoog (19 valsmeldingen bij een detectie van 50%) terwijl deze bij detectie in drie stappen daalt naar 1 valsmelding en het percentage gedetecteerde incidenten gelijk blijft. Hieruit kan men concluderen dat alle drie de stappen uitgevoerd moeten worden om incidenten met voldoende zekerheid te kunnen detecteren. Parametertest 3 in tabel 7.9 geeft de beste resultaten (50% gedetecteerd, 1 valsmelding) en deze parameters worden dan ook gebruikt voor de test van dit algoritme.

7.3.3. Implementatie McMaster algoritme

In het kader van dit afstudeeronderzoek is ook het McMaster algoritme binnen MoniGraph in Matlab geïmplementeerd. In bijlage D is de code van dit algoritme toegevoegd. Het verifiëren van de code is op de dezelfde wijze uitgevoerd als bij de andere methoden. Met behulp van breakpoints is de code doorgelopen om de juistheid te bepalen.

Bij de implementatie van dit algoritme zijn een aantal vereenvoudigen aangenomen, zo is de kritische capaciteit bij de test als een vaste waarde per rijstrook aangenomen. In werkelijkheid is deze kritische capaciteit afhankelijk van de locatie en situatie op dat moment. Ook is het algoritme niet toegepast indien de intensiteiten lager waren dan 750 vtg/uur, aangezien in dat geval in eerste instantie erg veel valsmeldingen werden gegeven. Bij lage intensiteiten werd immers in veel gevallen verkeersstoestand 2 weergegeven.

Voor het kalibreren zijn ook voor het McMaster algoritme een aantal parameters getest om een zo goed mogelijke instelling te krijgen. Ook bij dit algoritme zijn de parameters die te gevoelig of juist te ongevoelig stonden ingesteld aangepast. In onderstaande tabel zijn de onderzochte parameters weergegeven.

Tabel 7.10 Gebruikte parameters voor de kalibratie van het McMaster algoritme

	Paratest 1	Paratest 2	Paratest 3	Paratest 4
Vcrit (per strook)	1250	1000	1250	1000
K	0,80	0,80	0,70	0,70
min intensiteit	750	750	750	750

Voor alle vier de parameterinstellingen is voor de periode van 1 juni 2006 tot 15 juni 2006 het aantal gemiste meldingen, de betrouwbaarheid en de detectietijd bepaald. In tabel 7.11 zijn deze resultaten weergegeven.

Tabel 7.11 Resultaten parameter tests McMaster algoritme

	para1		para2		Para3		para4	
	Aantal	%	aantal	%	aantal	%	aantal	%
Gedetecteerde meldingen (totaal 6 incidenten)	6	100	6	100	6	100	6	100
Betrouwbaarheid (aantal valse meldingen)	1159		1007		471		369	
Detectietijd (aantal minuten na melding)	-5		-4,7		-3,3		-2,2	

Uit deze resultaten kan geconcludeerd worden dat dit Algoritme alle incident detecteert. Bij parameter test 3 en 4 zijn het aantal valsmeldingen beduidend minder dan bij de eerste twee test. De beste resultaten zijn te vinden bij de parameters van paratest 4. Er worden evenveel incidenten gedetecteerd als bij paratest 3 (100%), maar het aantal valsmeldingen is minder (369 tegenover 471). De detectietijd is bij paratest 3 wel iets beter, maar het hoge aantal valsmeldingen bij deze test weegt hier toch zwaarder.

7.3.4. Implementatie Strook algoritme

Ook dit algoritme is binnen MoniGraph geïmplementeerd voor dit onderzoek. Voor de implementatie zijn een aantal vereenvoudigingen aangenomen. Voor het bepalen van de grenswaardes in uitgegaan dat er sprake is van congestie indien de snelheden lager zijn dan 75 km/h. Indien de snelheden hoger zijn dan 75 km/uur is er sprake van vrije afwikkeling. De grenswaardes voor het testen van het algoritme zijn bepaald aan de hand van zes achtereenvolgende incidentvrije dagen op de A12 links en worden voor een periode van twee maanden de gelijke waarde aangenomen.

Voor het bepalen van de parameters wordt uitgegaan van de bevinden van de analyse van de incidenten uit hoofdstuk 5. Het verifiëren van de code is op de dezelfde wijze uitgevoerd als bij de andere methoden. Met behulp van breakpoints is de code doorgelopen om de juistheid te bepalen. In onderstaande tabel zijn de parameters weergegeven zoals deze zijn gebruikt in het algoritme.

Tabel 7.12 Gebruikte parameters het Strook algoritme

	waarde	Omschrijving
par_min_q	545 vtg/uur	de minimale intensiteit per rijstrook waarvoor het algoritme werkt
par_max_q	1750 vtg/uur	dit is de maximale intensiteit per rijstrook waarvoor het algoritme werkt
K₁	0,25	veiligheid minimale grenswaarde
K₂	1,25	veiligheid maximale grenswaarde

7.4 Resultaten test

Nadat alle detectiealgoritmen binnen MoniGraph zijn geïmplementeerd, kan de test worden uitgevoerd. De test is uitgevoerd voor de maanden juni en juli op de A12 links. In tabel 7.13 zijn de resultaten weergegeven voor alle vier de detectiemethoden. Hierbij moeten de volgende kanttekeningen worden gemaakt:

- Voor een aantal dagen was niet voldoende data beschikbaar, deze dagen zijn niet meegenomen in deze test. In totaal is de test voor 48 dagen uitgevoerd.
- Het incident van 23 juni is in hoofdstuk 5 geanalyseerd. Hieruit bleek dat dit incident eerder plaatsgevonden moet hebben dan de melding die door VCNL is gegeven. Ook bij twee van de vier detectiemethoden werd dit incident eerder gemeld (de andere methoden gaf geen melding). Voor dit incident is het tijdstip van melding dan ook veranderd naar minuut 710.
- Voor het incident van 27 juni op hmp 37,60 op minuut 488 was bij het bestuderen van de meetgegevens opgemerkt dat hier ook bij 391 een incident zichtbaar was. Door drie van de vier methoden is een melding gegeven op dit tijdstip en niet op het tijdstip 488. Om deze reden is ook bij deze melding het tijdstip veranderd.
- Voor het Strook algoritme is alleen naar de meldingen gekeken die zijn gegeven in het geval dat de snelheden groter waren dan 75 km/uur.
- Voor het Strook algoritme werden in eerste instantie veel incident niet gedetecteerd. Dit kwam doordat bij lage rijbaanintensiteiten de grenswaarde vermenigvuldigd met de k-waarde boven de 1,0 uitkomt. Om ervoor te zorgen dat in dat geval alsnog een melding wordt gegeven, is een voorwaarde aan het algoritme toegevoegd: indien het aandeel verkeer op een rijstrook gelijk is aan 1 en de grenswaarde voor deze rijbaanintensiteit (zonder k waarde) niet gelijk is aan 1 wordt er een melding gegeven.
- Daarnaast viel bij het bestuderen van de meldingen van het Strook algoritme op dat veel van de valsmeldingen werden gegeven op locaties waar de rijbaan uit vier rijstroken bestaat. Bij de resultaten is daarom bij het strookalgoritme onderscheidt gemaakt in het totale aantal valsmeldingen (Strook algoritme resultaten totaal) en het aantal valsmeldingen waarbij de rijbaan uit minder dan vier rijstroken bestaat (Strook algoritme resultaten minder dan 4 rijstroken).

Tabel 7.13 Resultaten test detectiealgoritmes op de A12 links voor de maanden juni en juli

	Blokkade detector	Presikhaaf algoritme	McMaster algoritme	Strook algoritme (resultaten totaal)	Strook algoritme (resultaten minder dan 4 rijstroken)
Betrouwbaarheid (aantal valsmeldingen)	25	1	821	703	109
Volledigheid (% gedetecteerd)	50,00 %	18,75 %	100 %	62,50 %	
Detectie tijd (minuten na melding)	-1,63	-13,3	0,13		-7,5

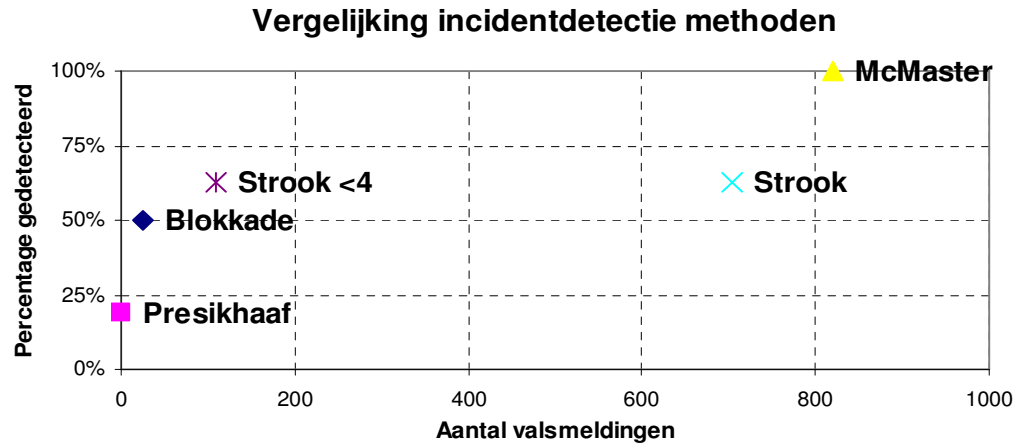
Uit de resultaten in bovenstaande tabel blijkt dat de blokkadedetector de helft van de incidenten heeft gedetecteerd. Daarbij zijn er 25 valse meldingen gegeven. De detectietijd is in vergelijking met de andere methodes niet erg hoog. Een nadeel van de Blokkadedetector is dat er veel parameters ingesteld moeten worden, waardoor de complexiteit van het algoritme toeneemt. Het Presikhaaf algoritme scoort vooral zeer goed op betrouwbaarheid. In de testperiode is maar 1 foutmelding gegeven. Daar tegenover is wel maar 18,75 procent van de incidenten gedetecteerd. Dit algoritme kan dus worden gezien als een zeer betrouwbaar algoritme, maar mist ook veel incidenten. Ook bij dit algoritme moeten veel parameters ingesteld worden wat de methode complexer maakt. Het McMaster algoritme laat juist het tegenoverstelde zien van het Presikhaaf algoritme. Alle incidenten zijn door het algoritme gedetecteerd, maar het aantal valsmeldingen bedraagt 821. Met de parameterinstellingen die bij deze test zijn gebruikt kan worden geconcludeerd dat dit algoritme te onbetrouwbaar is om in te zetten. Ook op het gebied van detectietijd scoort dit algoritme het slechts. Door nader onderzoek naar dit algoritme (parameterinstellingen, bepalen bezettingsgraad) te verrichten kan het aantal valsmeldingen wellicht omlaag worden gebracht (waarschijnlijk zal dan ook een lager percentage van de incidenten gedetecteerd worden). Het voordeel van dit algoritme is wel dat de parameters in principe niet van te voren hoeven te worden ingesteld, maar online bepaald kunnen worden. Hierdoor is dit algoritme breed inzetbaar. Het Strook algoritme detecteert de incidenten in het algemeen in een vroeg stadium. In totaal worden voor dit algoritme 703 valsmeldingen gegeven. Dit aantal daalt echter aanzienlijk indien alleen naar de valse meldingen wordt gekeken bij rijbanen die uit minder dan 4 stroken bestaan. Dit algoritme is dan ook zeker nog niet optimaal en nader onderzoek is noodzakelijk om dit algoritme praktisch te kunnen toepassen.

7.5 Vergelijkingen en conclusies

Om duidelijker inzicht in de prestaties van de verschillende methodes te geven, is in onderstaande figuur het aantal valsmeldingen uitgezet tegen het percentage gedetecteerde incidenten. In deze grafiek is duidelijk zichtbaar dat naarmate een detectiealgoritme meer incidenten detecteert ook het aantal valsmeldingen toenemen.

Figuur 7.1:

Grafiek van het aantal valse meldingen uitgezet tegen het percentage gedetecteerde incidenten voor alle detectiemethoden



Gezien de resultaten van deze test kan worden gezegd dat het McMaster algoritme geen goede resultaten heeft. Hoewel dit algoritme alle incidenten detecteert, is het aantal foutmeldingen van 821 onacceptabel en maakt het algoritme onbetrouwbaar. Ook detecteert het algoritme de incidenten in een laat stadium. Door nader onderzoek kan dit algoritme wellicht geoptimaliseerd worden. In het volgende hoofdstuk worden enkele aanbevelingen gegeven voor nader onderzoek.

Ook de resultaten van het Stroom algoritme (voor alle locaties) zijn niet goed. Gemiddeld worden meer dan 14,5 valsmeldingen per dag gegeven, waardoor het algoritme te onbetrouwbaar wordt. Indien alleen wordt gekeken naar het aantal valse meldingen op locaties met minder dan vier rijstroken wordt dit aantal aanzienlijk minder. In dat geval worden nog maar gemiddeld 2,3 valsmeldingen per dag gegeven. Dit aantal is echter nog te hoog om dit algoritme betrouwbaar te noemen. In deze test zijn echter voor dit algoritme enkele vereenvoudigingen aangenomen. Door nader onderzoek zal dit algoritme zeker beter kunnen presteren. Een voordeel van een incident detectie methode op rijstrokniveau is wel dat er meer informatie uit een incident melding gehaald kan worden, onder andere over de rijstrook waarop zich de problemen voordoen. In het volgende hoofdstuk worden de aanbevelingen voor nader onderzoek naar dit algoritme besproken.

Zowel de blokkadedetector als het Presikhaaf algoritme hebben een laag aantal valse meldingen, waardoor de betrouwbaarheid en ook de praktische inzetbaarheid van deze methoden vergroot. De blokkadedetector geeft gemiddeld eens in de twee dagen een valsmelding en het Presikhaaf algoritme over de gehele testperiode in totaal maar 1 valsmelding. Deze aantallen zijn acceptabel om praktische toe te passen. De detectiepercentage van het Presikhaaf algoritme is met 18,75% laag. De blokkadedetector detecteert de helft van de incidenten. Gelet op de detectietijd is voornamelijk het Presikhaaf algoritme snel. Door de hoge betrouwbaarheid en snelle detectie van het Presikhaaf algoritme zal dit algoritme goed kunnen functioneren in samenwerking met bijvoorbeeld de Blokkadedetector. als algoritme op zich detecteert het Presikhaaf algoritme te weinig incidenten (18,75%)

8. Conclusies en aanbevelingen

Het doel van dit onderzoek was ten eerste om een antwoord te krijgen op de vraag of gebruik van rijstrookdata bruikbaar kan zijn voor het detecteren van incidenten. Hierbij is gekeken of tijdens incidenten bijzondere rijstrookverdelingen waar te nemen zijn die karakteristiek zijn voor incidenten. Op basis van de resultaten van de analyse van enkele incidenten is een eerste opzet gemaakt voor een detectiemethode. Daarnaast was er de vraag of andere ontwikkelde methoden in Nederland toegepast zouden kunnen worden. Hiervoor zijn naast de in Nederland operationele Blokkadedetector, het Presikhaaf algoritme en het McMaster algoritme nader bekeken. Deze algoritmes zijn samen met het nieuw ontwikkelde algoritme getest. In dit hoofdstuk worden de belangrijkste bevindingen en conclusie van dit onderzoek weergegeven en worden aanbevelingen gegeven voor vervolgonderzoek.

8.1 Conclusies

Op grond van de analyse van de incidenten op strookniveau kunnen de volgende conclusies worden getrokken:

- Hoe het verkeer zich verdeelt over de rijstroken is afhankelijk van de rijbaanintensiteit. Afhankelijk van de locatie is er onder 'normale' omstandigheden een vast patroon zichtbaar in het rijstrookgebruik. Over het algemeen geldt dat bij lage rijbaanintensiteiten het grootste aandeel van het verkeer op de rechter rijstrook of rijstroken rijdt. Als de rijbaanintensiteiten toenemen, verdeelt het verkeer zich gelijkmatiger over de rijstroken. Bij hoge rijbaanintensiteiten wordt het aandeel verkeer op de linker rijstroken steeds hoger (of in het geval van twee rijstroken zelfs beduidend hoger dan op de rechter rijstrook). Voor een incidentvrije situatie zijn dus grenswaardes vast te stellen voor de verdeling van het verkeer over de rijstroken
- In het geval dat een incident heeft plaatsgevonden is een duidelijke afwijking zichtbaar in de verdeling van het verkeer over de rijstroken. De rijbaanintensiteiten moeten dan wel hoog genoeg zijn.
- De afwijkende rijstrookverdeling die zich voordoet in het geval van een incident is in bepaalde intensiteitsklassen uitzonderlijk genoeg om als incidenten te kunnen detecteren. Bij zeer lage rijbaanintensiteiten (ongeveer lager dan 500 vtg/uur per rijstrook) is het geen uitzonderlijke situatie dat plotseling veel voertuigen gebruik maken van een andere rijstrook. Ook bij zeer hoge rijbaanintensiteiten is het lastig vast te stellen of een bepaalde rijstrookverdeling uitzonderlijk genoeg is.
- Het opstellen van aparte grenswaardes voor vrije afwikkeling en congestie is noodzakelijk om goede grenswaardes vast te stellen die bruikbaar zijn om incidenten te detecteren.

Op basis van de resultaten uit de analyse van de meetgegevens op

rijstrookniveau tijdens incidenten is een voorstel gegeven voor een detectiealgoritme op rijstrookniveau. Deze Strook methode zoekt naar afwijkende strookverdeling. Indien op een bepaalde locatie het aandeel verkeer te veel afwijkt van de minimale en maximale grenswaardes van een 'normale' situatie voor de desbetreffende rijbaanintensiteit, wordt een melding gegeven van een incident. Deze grenswaardes worden per rijbaanintensiteit bepaald uit de minimale en maximale gemeten rijstrookaandelen over een aantal incident vrije dagen. Zoals uit de conclusies van de analyse van de incidenten naar voren komt, kan er voor bepaalde intensiteitsklassen niet van een uitzonderlijke situatie worden gesproken. Om deze reden worden lage rijbaanintensiteiten (ongeveer 545 vtg/uur per rijstrook) en zeer hoge rijbaanintensiteiten (ongeveer 1750 vtg/uur per rijstrook) uitgesloten voor detectie.

Deze nieuwe methode op rijstrookniveau is samen met drie andere nader onderzochte methodes, de Blokkadedetector, Presikhaaf algoritme en het McMaster algoritme getest. Dit is gedaan voor de maanden juni en juli op de A12. De data van de incidentmeldingen zijn afkomstig van VCNL (Verkeerscentrale Nederland).

De prestaties van de verschillende methoden lopen erg uiteen. Op het McMaster algoritme na detecteren de methoden de incidenten gemiddeld eerder dan de gerapporteerde incidenten van VCNL. Dit houdt dus in dat automatische incident detectie een toegevoegde waarde heeft op het gebied van incidentdetectie.

De resultaten uit de test geven weer dat het Presikhaaf algoritme de incidenten zeer snel detecteert en zeer betrouwbaar is (1 valsmelding). Daartegenover heeft het algoritme een laag percentage (18,75%) van het aantal incidenten gedetecteerd. Het McMaster algoritme geeft juist tegenovergestelde resultaten weer. Deze methode heeft alle incidenten gedetecteerd, maar is ook zeer onbetrouwbaar (821 valsmeldingen). Ook scoort dit algoritme het slechts met betrekking tot de detectietijd. De blokkadedetector scoort gemiddeld met 25 valsmeldingen en een detectie van de helft van de incidenten. Het Strook algoritme geeft voornamelijk valsmeldingen op rijbanen met veel rijstroken (703 valsmeldingen in totaal waarvan 109 valse meldingen op rijbanen met minder dan vier rijstroken). Over het algemeen detecteert het algoritme de incidenten redelijk snel en is 62,5 procent van de incidenten gedetecteerd.

Op basis van de resultaten van deze test kan worden geconcludeerd dat zowel het McMaster algoritme als het Strook algoritme (voor alle locaties) slechte resultaten geven. Beide methodes detecteren een hoog percentage incidenten (het McMaster algoritme zelfs alle incidenten), maar het aantal valse meldingen is ook zeer hoog (beide methoden meer dan 14 per dag). Door deze hoge aantallen valsmeldingen zijn deze methodes niet acceptabel voor praktische toepassing. Bij het Strook algoritme daalt het aantal valsmeldingen aanzienlijk (van 703 naar 109) indien alleen naar de resultaten wordt gekeken van locaties met minder dan vier rijstroken. Een totaal van 109 valsmeldingen (2,2 per dag) is weliswaar nog niet acceptabel voor praktische toepassing, maar dit algoritme functioneert zeker niet optimaal. Het aanscherpen van de parameterinstellingen en het wegnemen van de vereenvoudigingen bij deze methodes zal betere resultaten opleveren. Een voordeel van het Strook algoritme boven de andere algoritmes is dat dit algoritme meer informatie over het incident geeft, bijvoorbeeld welke strook of stroken het betreft. In de volgende paragraaf worden voor beide methoden aanbevelingen gegeven voor nader onderzoek die wellicht voor verbeteringen van deze methodes kunnen zorgen.

Het Presikhaaf algoritme en de Blokkadedetector leveren beide goede resultaten als wordt gekeken naar de betrouwbaarheid. Beide algoritmes hebben een laag aantal valse meldingen. De blokkadedetector geeft gemiddeld eens in de twee dagen een valsmelding en het Presikhaaf algoritme over de gehele testperiode in totaal maar 1 valsmelding. Deze aantallen zijn acceptabel

om praktische toe te passen. Aangezien het Presikhaaf algoritme zeer betrouwbaarheid is en een snelle detectietijd heeft, maar een laag percentage detecteert, zou dit algoritme voornamelijk toegevoegde waarde hebben om het naast de Blokkadedetector of andere detectiemethoden operationeel te hebben.

Kort kan worden gezegd dat alle methodes bewezen hebben, toepasbaar te zijn voor incident detectie. Echter detecteert het Presikhaaf algoritme een te laag percentage incidenten en zal dit algoritme alleen toegevoegde kunnen hebben door de snelle detectie als deze in combinatie met andere detectiealgoritmes werkzaam is. De Blokkadedetector geeft redelijke resultaten, maar de resultaten zijn niet optimaal en laten nog veel ruimte voor verbeteringen. Het McMaster algoritme geeft in de tests geen goede resultaten, maar door het wegnemen van vereenvoudigingen en beter instellingen van parameters zullen de resultaten van dit algoritme wellicht verbeterd worden. Aangezien dit algoritme alle incidenten detecteert wordt nader onderzoek naar dit algoritme dan ook aanbevolen. Ook voor het Strookalgoritme wordt nader onderzoek aanbevolen. De resultaten van de tests zijn niet optimaal, voornamelijk op locaties waar de rijbaan uit meer dan drie rijstroken bestaat. Het Strook algoritme zal zeker beter kunnen gaan presteren als vereenvoudigen en onzekerheden weggenomen worden. Aanbevelingen voor nader onderzoek worden in de volgende paragraaf beschreven.

8.2 Aanbevelingen voor vervolgonderzoek

Uit dit onderzoek kan geconcludeerd worden dat verkeersgegevens op rijstrookniveau bruikbaar zijn voor het detecteren van incidenten. Echter de voorgestelde detectiemethode op rijstrookniveau is mede wegens beperkingen in tijd die voor dit afstudeeronderzoek staat, niet optimaal. Hiervoor is nader onderzoek zeker noodzakelijk. Hetzelfde geldt voor de andere onderzochte detectiemethoden. Hieronder worden eerst enkele aanbevelingen gegeven voor nader onderzoek voor de methode op rijstrookniveau.

- Het eerste punt waar nader onderzoek nodig is, is de instellingen van de parameters. In dit onderzoek zijn de parameters voor alle rijstroken en locaties (op het aantal rijstroken per rijbaan na) gelijk aangenomen. De werking van het algoritme zal te optimaliseren zijn door de parameters locatie en eventueel ook strookafhankelijk te maken.
- Verder onderzoek naar verschillen tussen rijbanen en is gewenst om het Strook algoritme beter te kunnen laten functioneren.
- Een ander punt waar nog aandacht aan besteed kan worden is het onderscheid tussen het vaststellen van de grenswaardes van situaties waarin sprake is van vrije afwikkeling en congestie. In dit onderzoek zijn de testresultaten van lage snelheden niet meegenomen. Om hier wat meer over te zeggen dient nader onderzoek te worden verricht
- Ook een verdere validatie op een groter traject en meer incidenten is een belangrijk aandachtspunt (dit geldt overigens ook voor de andere detectiemethoden).
- Om het aantal valsmeldingen te verkleinen kan een indeling in grotere intensiteitsklassen (bijvoorbeeld klassen van 500) of gebruik maken van afgevlakte meetgegevens (5-minuut gemiddelden) opties zijn. In beide gevallen wordt het algoritme minder 'gevoelig'. Fouten in de meetgegevens of door het grillige karakter van de verkeersstroom worden dan niet meteen als incidenten gedetecteerd.

Ten aanzien van de andere onderzochte en geteste detectiemethoden is ook nader onderzoek zeker gewenst. Aanvullend onderzoek zal hier met name liggen in de instelling van de parameters. Voor het McMaster algoritme zijn een aantal parameters locatieafhankelijk die in dit onderzoek niet locatieafhankelijk zijn aangenomen (V_{crit} , OC_{MAX}). Het nauwkeurig vaststellen van deze parameters kan de werking van het McMaster algoritme wellicht verbeteren. Ook het online bepalen van de parameters, aan de hand van recente meetgegevens kan de resultaten van dit algoritme verbeteren. Naast de parameterbepaling kan dit algoritme ook verder geoptimaliseerd worden door de bezettingsgraad nauwkeurig te bepalen. In dit onderzoek is bij het bepalen van deze grootte uitgegaan van een uniforme verkeersstroom. In werkelijkheid is dit niet het geval. Aangezien de detectielussen steeds verder ontwikkeld worden, zijn er op steeds meer locaties ook gegevens beschikbaar over de voertuigcategorieën. Door gebruik te maken van deze gegevens, kan de bezettingsgraad nauwkeuriger bepaald kunnen worden. Een andere mogelijkheid voor het McMaster algoritme is gebruik te maken van de meetgegevens die door de detectielussen in Nederland worden gemeten: de intensiteit en de snelheid. Het zelfde principe van het McMaster algoritme kan worden toegepast met het verschil dat er gebruik wordt gemaakt van de intensiteit en snelheid.

Een laatste punt voor een mogelijk nader onderzoek is onderzoek te verrichten naar de mogelijkheden om meerdere algoritmes naast elkaar operationeel te laten zijn en hier een beslissingsmodel aan te koppelen (Sheu en Ritchy (2000), Levin et al. (1979) en Cohen en Ketselidou (1993)). Het voordeel hiervan kan zijn minder vals meldingen worden gegeven en tevens bestaat de mogelijkheid dat meer incidenten worden gedetecteerd (verschillende detectiemethoden detecteren zoeken naar verschillende verkeerskenmerken van incidenten en detecteren dus ook verschillende incidenten).

Referenties

- [1] Transpute (2002), Onderzoek Modificatie AID – Blokkade detector, Knibbe W.J.J., Alkim T.K., Rijkswaterstaat, Adviesdienst Verkeer en Vervoer
- [2] Gall A.I. and Hall F.L. (1989), 'Distinguishing between incident congestion and recurrent congestion: A proposed logic', Transportation Research Record 1232, TRB, pp. 1-8
- [3] Hall F.L., Shi Y. and Atala G. (1993), 'On-line testing of the McMaster incident detection algorithm under recurrent congestion', Transportation Research Record 1394, pp.1-7
- [4] Rakha H., Hellinga B and Van Aerde M. (2004), 'Testbed for Evaluating Automatic Incident Detection Algorithms', Proceedings of the Intelligent Transportation Systems Safety and Security Conference
- [5] Martin P. T., Perrin J., Hansen B., Kump R. and Moore D. (2001), 'Incident Detection Algorithm Evaluation', University of Utah Prepared for Utah Department of Transportation
- [6] Hoogendoorn S.P., Botma H. and Minderhoud M.M. (2005), Traffic Flow Theory and Simulation, TU Delft, Faculty of Civil Engineering and Geosciences
- [7] Hoogendoorn, S.P., Ossen S.J.L. (2006), Hiaatdetectie, Technische Universiteit Delft Faculteit Civiele Techniek en Geowetenschappen
- [8] Ruiter de J.C.C., Schouten W.J.J.P, Frijdal J.H. (2002), Berekenen van reistijden door het MoniBas systeem, Ministerie van Verkeer en Waterstaat, Rijkswaterstaat (AVV) en Belmont Innovatie & Management, Haarlem, RTIC conferentie van het IEE
- [9] Hall, F.L. (1975), Traffic stream characteristics, update and expansion of the Transportation Research Board (TRB) Special Report 165, "*Traffic Flow Theory*," published in 1975.
- [10] Knibbe W.J.J. (2006), BRINTA, Automatic Incident Detection Algorithm – First principles & strategy for Java implementation
- [11] Knibbe W.J.J. (2005), Description of Blocking-the-Road Incident detection by Transputes Algorithm (BRINTA)
- [12] Middelham F.(2005), System/Subsystem Specification Blokkadedetector V2

www.intraffice.nl/projecten/blokkadedetector/

<http://www.mcmaster.ca/graduate/flhall/homepage>

Bijlage A Overzicht incidentmeldingen van VCNL

Het VCNL rapporteert op landelijk niveau allerlei gegevens over de verkeerssituatie op de Nederlandse wegen. Indien er geen sprake is van een normale doorstroming van het verkeer, wordt dit gerapporteerd. Onder andere wordt informatie over de datum, locatie, begin en eindtijdstip, de oorzaak en filelengte beschreven. Bij de beschrijving van de oorzaak van de congestie wordt onder andere onderscheidt gemaakt tussen dagelijkse ochtend en avondspitsen, bijzondere situaties en incidenten. Aan de hand hiervan zijn alle incidenten uit deze lijst te filteren. In onderstaande tabellen zijn alle incidentmelding voor de A12 en A20 in de maanden juni en juli weergegeven¹.

**Tabel A.1 incidentmeldingen A12 district Zuid-Holland
maand juni en juli van VCNL**

DTFileBe gin	RO UT EO MS	Rij ba an	Hecto meter aand uiding	Begin uur	Eind uur	beg in minu ut	eind minuu t	Oorzaak	Oorzaak _2	Incidentt ypeCL1	Weekd ag	So mF ile Le ngt e	So mFi leD uur	Som FileZ waar te
5-6	A12	Li	16,60	11:43	13:15	704	796	verkeers ongeval	ongeval	Overig	maand ag	4	93	372
5-6	A12	Li	13,10	13:34	14:07	815	848	verkeers ongeval	ongeval	Overig	maand ag	3,4	34	116
6-6	A12	Li	13,10	17:02	17:37	1023	1058	kijkfile		Ongeval	dinsda g	4,4	36	159
6-6	A12	Li	27,00	18:49	19:01	1130	1142	verkeers ongeval	ongeval	Overig	dinsda g	2,8	13	36
12-6	A12	Li	25,40	16:43	16:50	1004	1011	gestrand voertuig	pech	gestrand voertuig	maand ag	2	8	16
14-6	A12	Li	25,40	16:39	16:42	1000	1003	verkeers ongeval	ongeval	Overig	woens dag	3	4	12
14-6	A12	Li	29,60	16:44	18:09	1005	1090	overige oorzaken	modder of olie op weg	Lading	woens dag	15	86	1293
15-6	A12	Li	8,50	8:59	10:01	540	602	verkeers ongeval	ongeval	Overig	donder dag	7,2	63	455
20-6	A12	Li	4,40	7:42	11:00	463	661	verkeers ongeval	ongeval	Overig	dinsda g	5,8	199	1160
20-6	A12	Li	29,60	8:27	8:35	508	516	verkeers ongeval	ongeval	Overig	dinsda g	2,4	9	22
22-6	A12	Li	37,60	8:21	9:15	502	556	verkeers ongeval	ongeval	Overig	donder dag	4,6	55	252

¹ De hier weergegeven gegevens is bewerkt en alleen de voor dit onderzoek relevante gegevens zijn weergegeven

23-6	A12	Li	26,70	12:33	12:57	754	778	verkeers ongeval	ongeval	Overig	vrijdag	2,7	25	68
27-6	A12	Li	6,60	7:52	7:58	473	479	kijkfile		Ongeval	dinsda g	8,9	7	62
27-6	A12	Li	37,60	8:07	8:13	488	494	gestrand voertuig	pech	gestrand voertuig	dinsda g	4	7	28
27-6	A12	Li	28,00	8:45	9:58	526	599	verkeers ongeval	ongeval	Overig	dinsda g	6,7	74	493
30-6	A12	Li	13,10	8:48	9:19	529	560	verkeers ongeval	ongeval	Overig	vrijdag	4,6	32	146
30-6	A12	Li	19,50	9:20	9:36	561	577	verkeers ongeval	ongeval	Overig	vrijdag	2,7	17	46
2-7	A12	Li	22,30	10:29	11:37	630	698	verkeers ongeval	ongeval	Overig	zonda g	4,1	69	282
6-7	A12	Li	27,00	17:24	18:06	1045	1087	verkeers ongeval	ongeval	Overig	donder dag	5,2	43	225
6-7	A12	Li	29,60	18:07	18:49	1088	1130	verkeers ongeval	ongeval	Overig	donder dag	4,4	43	188
7-7	A12	Li	27,00	17:58	19:31	1079	1172	verkeers ongeval	ongeval	Overig	vrijdag	10, 8	94	1012
7-7	A12	Li	37,60	19:22	20:17	1163	1218	verkeers ongeval	ongeval	Overig	vrijdag	4,8	56	269
8-7	A12	Li	8,50	13:45	15:07	826	908	verkeers ongeval	ongeval	Overig	zaterd ag	7,2	83	596
29-7	A12	Li	29,60	14:00	14:31	841	872	verkeers ongeval	ongeval	Overig	zaterd ag	3	32	95
29-7	A12	Li	29,60	14:35	14:43	876	884	verkeers ongeval	ongeval	Overig	zaterd ag	2	9	18
2-6	A12	Re	22,30	17:01	17:12	1022	1033	gestrand voertuig	pech	gestrand voertuig	vrijdag	2,3	12	28
6-6	A12	Re	37,60	15:56	17:19	957	1040	verkeers ongeval	ongeval	Overig	dinsda g	8	84	672
6-6	A12	Re	14,70	16:42	17:18	1003	1039	verkeers ongeval	ongeval	Overig	dinsda g	5	37	184
6-6	A12	Re	14,70	17:20	18:22	1041	1103	verkeers ongeval	ongeval	Overig	dinsda g	6	63	378
7-6	A12	Re	37,60	15:46	17:07	947	1028	verkeers ongeval	ongeval	Overig	woens dag	8,5	82	701
7-6	A12	Re	31,90	17:08	17:10	1029	1031	verkeers ongeval	ongeval	Overig	woens dag	10	3	30
8-6	A12	Re	19,50	13:59	15:02	840	903	verkeers ongeval	ongeval	Overig	donder dag	3,1	64	197
8-6	A12	Re	3,30	17:41	18:26	1062	1107	verkeers ongeval	ongeval	Overig	donder dag	2	46	92
9-6	A12	Re	4,40	0:00	0:50	1	51	verkeers ongeval	ongeval vracht auto	Ongeval	vrijdag	2,9	51	149
9-6	A12	Re	6,10	13:21	13:22	802	803	verkeers ongeval	ongeval	Overig	vrijdag	2	2	4
10-6	A12	Re	37,60	23:24	23:59	1405	1440	verkeers ongeval	ongeval	Overig	zaterd ag	3,1	36	111

11-6	A12	Re	37,60	0:00	1:29	1	90	verkeers ongeval	ongeval	Overig	zonda g	4,6	90	417
12-6	A12	Re	11,70	18:31	19:25	1112	1166	verkeers ongeval	ongeval	Overig	maand ag	3,2	55	174
13-6	A12	Re	31,90	19:25	19:39	1166	1180	verkeers ongeval	ongeval	Overig	dinsda g	3	15	45
20-6	A12	Re	20,50	15:50	16:06	951	967	gestrand voertuig	pech	gestrand voertuig	dinsda g	2	17	34
20-6	A12	Re	20,50	16:07	17:42	968	1063	verkeers ongeval	ongeval	Overig	dinsda g	5,9	96	569
21-6	A12	Re	31,90	8:33	9:40	514	581	verkeers ongeval	ongeval	Overig	woens dag	3,3	68	221
23-6	A12	Re	37,60	14:52	15:01	893	902	verkeers ongeval	ongeval	Overig	vrijdag	3,5	10	35
23-6	A12	Re	37,60	15:02	17:07	903	1028	verkeers ongeval	ongeval	Overig	vrijdag	8,7	126	1099
23-6	A12	Re	37,60	15:24	15:25	925	926	verkeers ongeval	ongeval	Overig	vrijdag	3	2	6
28-6	A12	Re	17,80	15:35	17:07	936	1028	verkeers ongeval	ongeval	Overig	woens dag	5	93	463
5-7	A12	Re	31,90	13:15	13:25	796	806	gestrand voertuig	pech	gestrand voertuig	woens dag	2	11	22
6-7	A12	Re	11,70	15:51	17:40	952	1061	verkeers ongeval	ongeval	Overig	donder dag	6	110	658
10-7	A12	Re	19,50	13:08	13:27	789	808	gestrand voertuig	pech	gestrand voertuig	maand ag	2	20	40
10-7	A12	Re	19,50	13:30	14:06	811	847	gestrand voertuig	pech	gestrand voertuig	maand ag	3	37	111
12-7	A12	Re	11,70	17:26	19:07	1047	1148	verkeers ongeval	ongeval	Overig	woens dag	4,1	102	420
13-7	A12	Re	4,40	7:48	7:51	469	472	verkeers ongeval	ongeval	Overig	donder dag	2	4	8
13-7	A12	Re	4,40	7:53	8:23	474	504	verkeers ongeval	ongeval	Overig	donder dag	2	31	62
13-7	A12	Re	37,60	19:07	19:52	1148	1193	verkeers ongeval	ongeval	Overig	donder dag	2,1	46	96
18-7	A12	Re	37,60	15:39	15:47	940	948	overige oorzaken	overig	Overig	dinsda g	3	9	27
26-7	A12	Re	31,90	17:39	19:09	1060	1150	verkeers ongeval	ongeval	Overig	woens dag	5,1	91	461
1-8	A12	Re	4,40	7:48	8:12	469	493	verkeers ongeval	ongeval	Overig	dinsda g	2	25	50

**Tabel A.1 incidentmeldingen A20 district Zuid-Holland
maand juni en juli van VCNL**

DTFik Begin	RO UT EO MS	Rij ba an	Hecto meter aandui ding	Begin uur	Ein d uur	begin minu ut	eind minuut	Oorzaak	Oorzaak _2	Incidentt ypeCL1	Weekd ag	File Len gte	File Duu r	File Zwaa rte
5-6	A2 0	Li	31,50	17:06	18: 47	1027	1128	verkeers ongeval	ongeval	Overig	maand ag	3	102	307
7-6	A2 0	Li	29,10	10:24	10: 25	625	626	verkeers ongeval	ongeval met vrachta uto	Ongeval	woens dag	2	2	4
8-6	A2 0	Li	44,40	6:57	8:0 9	418	490	verkeers ongeval	ongeval	Overig	donder dag	3,9	73	288
8-6	A2 0	Li	27,90	13:39	15: 49	820	950	verkeers ongeval	ongeval	Overig	donder dag	3,9	131	517
9-6	A2 0	Li	23,80	17:36	18: 52	1057	1133	verkeers ongeval	ongeval	Overig	vrijdag	2,8	77	213
12-6	A2 0	Li	44,40	11:52	13: 56	713	837	gestrand voertuig	pech	gestrand voertuig	maand ag	3,9	125	488
12-6	A2 0	Li	17,20	17:45	18: 17	1066	1098	kijkfile		Ongeval	maand ag	2,9	33	96
15-6	A2 0	Li	27,90	13:59	16: 12	840	973	verkeers ongeval	ongeval	Overig	donder dag	4,1	134	547
19-6	A2 0	Li	32,20	6:37	8:5 2	398	533	verkeers ongeval	ongeval	Overig	maand ag	2	136	272
19-6	A2 0	Li	31,50	12:38	13: 16	759	797	kijkfile		Ongeval	maand ag	2,7	39	104
20-6	A2 0	Li	27,90	11:56	14: 11	717	852	verkeers ongeval	ongeval	Overig	dinsda g	5,1	136	687
21-6	A2 0	Li	26,70	12:37	14: 55	758	896	overige oorzaken	modder of olie op weg	Lading	woens dag	3,9	139	542
21-6	A2 0	Li	38,80	16:38	18: 36	999	1117	verkeers ongeval	ongeval	Overig	woens dag	4,4	119	518
22-6	A2 0	Li	29,10	12:26	13: 09	747	790	verkeers ongeval	ongeval	Overig	donder dag	3,7	44	161
25-6	A2 0	Li	31,50	18:12	18: 32	1093	1113	verkeers ongeval	ongeval	Overig	zonda g	2	21	42
30-6	A2 0	Li	14,70	6:37	6:5 0	398	411	verkeers ongeval	ongeval	Overig	vrijdag	2,2	14	31
1-7	A2 0	Li	44,40	17:02	19: 20	1023	1161	verkeers ongeval	ongeval	Overig	zaterd ag	4,4	139	614
2-7	A2 0	Li	11,30	11:52	11: 56	713	717	verkeers ongeval	ongeval	Overig	zonda g	2,6	5	13
2-7	A2 0	Li	11,30	11:57	13: 10	718	791	verkeers ongeval	ongeval	Overig	zonda g	4,1	74	302
2-7	A2 0	Li	11,30	12:56	13: 28	777	809	verkeers ongeval	ongeval	Overig	zonda g	2	33	66

4-7	A20	Li	29,10	9:56	13:35	597	816	verkeers ongeval	ongeval	Overig	dinsdag	4,9	220	1071
12-7	A20	Li	44,40	18:31	18:55	1112	1136	verkeers ongeval	ongeval	Overig	woensdag	3	25	74
13-7	A20	Li	27,90	9:21	11:03	562	664	intensiteit/capaciteit	afgesloten weg (deel)	Overig	donderdag	4,6	103	470
14-7	A20	Li	29,10	10:40	10	641	652	gestrand voertuig	pech	gestrand voertuig	vrijdag	2,2	12	26
17-7	A20	Li	31,50	11:03	11:41	664	702	verkeers ongeval	ongeval	Overig	maandag	2,9	39	112
18-7	A20	Li	17,20	16:37	17:14	998	1035	verkeers ongeval	ongeval	Overig	dinsdag	2,5	38	94
19-7	A20	Li	27,90	14:38	15:16	879	917	verkeers ongeval	ongeval	Overig	woensdag	2,1	39	81
20-7	A20	Li	27,90	8:21	8:57	502	538	verkeers ongeval	ongeval	Overig	donderdag	4,4	37	162
20-7	A20	Li	27,90	8:58	9:20	539	561	verkeers ongeval	ongeval	Overig	donderdag	2	23	46
26-7	A20	Li	29,10	17:03	18:35	1024	1116	verkeers ongeval	ongeval	Overig	woensdag	4,1	93	378
2-8	A20	Li	29,10	10:09	10:44	610	645	verkeers ongeval	ongeval	Overig	woensdag	2,6	36	94
4-8	A20	Li	11,30	9:24	9:36	565	577	verkeers ongeval	ongeval	Overig	vrijdag	2	13	26
6-6	A20	Re	29,10	8:35	9:54	516	595	verkeers ongeval	ongeval met vrachtauto	Ongeval	dinsdag	2,8	80	222
7-6	A20	Re	29,90	10:27	10:30	628	631	verkeers ongeval	ongeval met vrachtauto	Ongeval	woensdag	4	4	16
7-6	A20	Re	29,90	10:31	10:46	632	647	verkeers ongeval	ongeval met vrachtauto	Ongeval	woensdag	2,8	16	45
7-6	A20	Re	29,90	10:48	11:08	649	669	verkeers ongeval	ongeval met vrachtauto	Ongeval	woensdag	2,3	21	48
12-6	A20	Re	44,40	6:34	8:14	395	495	overige oorzaken	overig	Overig	maandag	5	101	501
12-6	A20	Re	20,50	17:44	18:17	1065	1098	verkeers ongeval	ongeval	Overig	maandag	3,2	34	110
12-6	A20	Re	20,50	18:18	18:21	1099	1102	verkeers ongeval	ongeval	Overig	maandag	2	4	8
15-6	A20	Re	29,90	8:27	8:34	508	515	kijkfile		Ongeval	donderdag	2	8	16
16-6	A20	Re	31,50	11:10	12:46	671	767	verkeers ongeval	ongeval	Overig	vrijdag	4,9	97	473
19-6	A20	Re	31,50	12:39	13:37	760	818	verkeers ongeval	ongeval	Overig	maandag	3,9	59	232

21-6	A2 0	Re	29,90	18:12	18: 22	1093	1103	verkeers ongeval	ongeval	Overig	woens dag	2	11	22
30-6	A2 0	Re	44,40	9:10	9:3 0	551	571	verkeers ongeval	ongeval	Overig	vrijdag	3	21	63
30-6	A2 0	Re	29,10	12:46	13: 40	767	821	verkeers ongeval	ongeval	Overig	vrijdag	2,2	55	121
1-7	A2 0	Re	42,20	16:37	21: 45	998	1306	verkeers ongeval	ongeval	Overig	zaterd ag	7,2	309	2227
4-7	A2 0	Re	29,10	12:02	13: 26	723	807	verkeers ongeval	ongeval	Overig	dinsda g	3,1	85	260
5-7	A2 0	Re	44,40	5:57	7:4 7	358	468	verkeers ongeval	ongeval	Overig	woens dag	5	111	559
8-7	A2 0	Re	42,20	15:02	18: 21	903	1102	verkeers ongeval	ongeval	Overig	zaterd ag	7,1	200	1420
8-7	A2 0	Re	42,20	15:59	15: 59	960	960	verkeers ongeval	ongeval	Overig	zaterd ag	4	1	4
13-7	A2 0	Re	27,90	9:20	9:2 0	561	561	intensitei t/capacit eit	afgeslot en weg (deel)	Overig	donder dag	2	1	2
13-7	A2 0	Re	29,90	16:03	18: 37	964	1118	verkeers ongeval	ongeval	Overig	donder dag	5	155	769
14-7	A2 0	Re	47,60	14:50	16: 51	891	1012	overige oorzaken	overig	Overig	vrijdag	4,7	122	578
17-7	A2 0	Re	44,40	17:58	18: 38	1079	1119	gestrand voertuig	pech	gestrand voertuig	maand ag	2,8	41	116
17-7	A2 0	Re	29,90	11:09	12: 03	670	724	verkeers ongeval	ongeval	Overig	maand ag	3,1	55	169
18-7	A2 0	Re	31,50	8:21	8:3 8	502	519	verkeers ongeval	ongeval	Overig	dinsda g	2,2	18	40
18-7	A2 0	Re	31,50	8:42	8:5 7	523	538	verkeers ongeval	ongeval	Overig	dinsda g	2,5	16	40
26-7	A2 0	Re	31,50	17:07	18: 31	1028	1112	verkeers ongeval	ongeval	Overig	woens dag	4,5	85	380
3-8	A2 0	Re	33,90	7:26	8:3 4	447	515	gestrand voertuig	pech	Gestran d voertuig	donder dag	3,3	69	231
4-8	A2 0	Re	14,70	18:18	18: 42	1099	1123	gestrand voertuig	pech	Gestran d voertuig	vrijdag	2	25	50
4-8	A2 0	Re	47,60	10:51	11: 54	652	715	verkeers ongeval	ongeval	Overig	vrijdag	4,2	64	266

Bijlage B Basis verkeersstroomtheorie

De drie basisgrootheden van de verkeersstroomtheorie

In de verkeersstroomtheorie zijn drie basisgrootheden van belang:

- Intensiteit q (aantal voertuigen per tijdseenheid)
- Dichtheid k (aantal voertuigen per kilometer weg)
- Snelheid u

Tussen deze basisgrootheden bestaat een fundamentele relatie die er op macroscopisch niveau als volgt uit ziet:

$$q = k \cdot u \quad \text{[B.1]}$$

waarin: q = intensiteit in vtg/uur
 k = dichtheid in vtg/km
 u = snelheid in km/uur

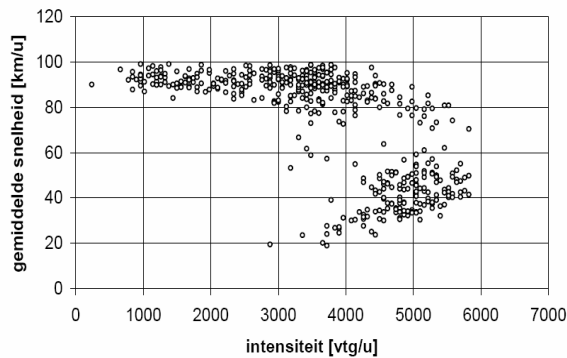
Door deze relatie zijn de intensiteit, dichtheid en gemiddelde snelheid onherroepelijk met elkaar verbonden. Doordat er maar twee onafhankelijke grootheden bestaan, leidt de derde altijd uit de andere twee.

Het fundamenteel diagram

Als we de drie basisgrootheden over een gehele dag elke minuut meten en deze in grafieken uitzetten, ontstaan onderstaande grafieken:

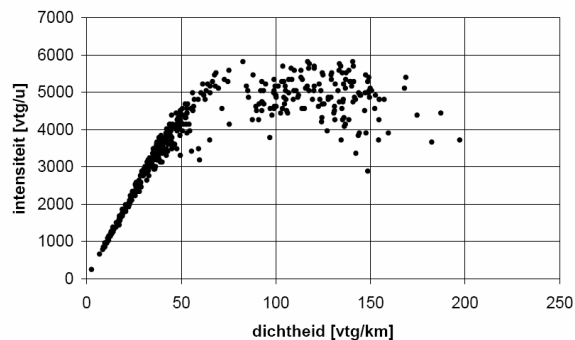
.....
Figuur B1:

Waarnemingspunten in het q-u diagram



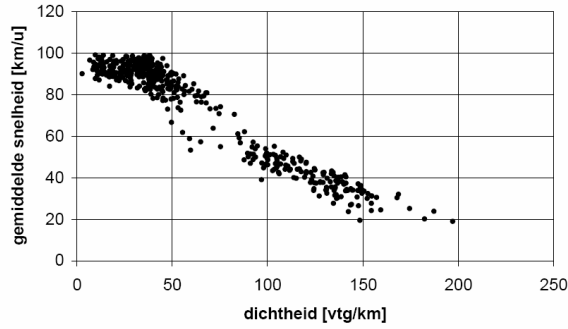
.....
Figuur B2:

Waarnemingspunten in het k-q diagram



.....
Figuur B3:

Waarnemingspunten in het q-u diagram

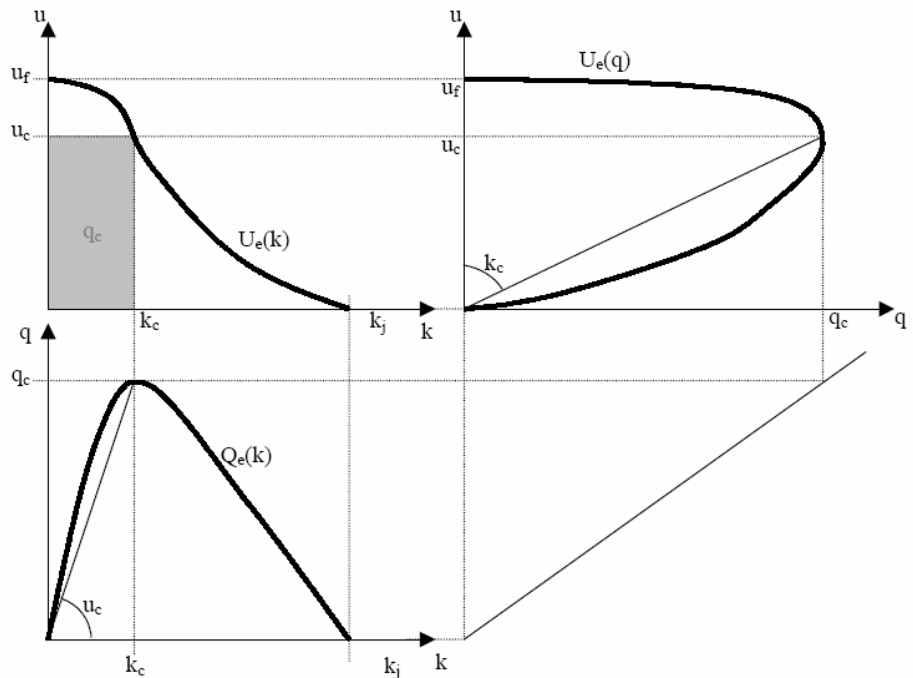


De waarnemen uit bovenstaande grafieken zijn verricht op een werkelijke snelweg waar verschillende voertuigtypen voorkomen en bestuurders variëren in gedrag. Ook is er in werkelijkheid geen sprake van een stationaire toestand doordat voertuigen accelereren en afremmen. Toch is in de grafieken van de waarnemingspunten al een duidelijk patroon zichtbaar.

Het verkeer op de weg bevindt zich altijd in een bepaalde toestand die gekenmerkt wordt door de intensiteit, snelheid en dichtheid. Als we uitgaan van een homogeen samengestelde verkeersstroom (alle voertuigen zijn gelijk) en van een stationaire situatie (verkeersstroom verandert niet over de tijd en weg) vormen deze drie grootheden de fundamentele diagrammen. Deze zijn in onderstaand figuur weergegeven.

.....
Figuur B4:

Fundamentele diagrammen in de verkeersstroomtheorie



Deze fundamentele diagrammen kunnen een goed inzicht geven over de kwaliteit van de verkeersafwikkeling op een bepaald wegvak.

In het fundamenteel diagram zijn twee afwikkelingsniveaus te onderscheiden: een vrije-afwikkelingstak en een congestietak. Het verkeer in de vrije-afwikkelingstoestand rijdt met een hoge snelheid bij een lage dichtheid. Als vervolgens de vraagintensiteit de capaciteit overschrijdt gaat de vrije-afwikkelingstoestand over in een congestiesituatie, waarin de snelheid laag is

en de dichtheid hoog (zie bovenstaand figuur).

In de fundamenteel diagrammen zijn een aantal punten belangrijk om te herkennen:

- k_j (vtg/km) = de stremmingsdichtheid: een dichtheid waarbij alle voertuigen met minimale afstand van elkaar stilstaan. De weg is verzadigd en de intensiteit en snelheden zijn dan nul.
- u_f (km/uur) = vrije snelheid van het verkeer (free flow): voertuigen hoeven hun snelheid niet aan andere voertuigen aan te passen, deze grootte is onder andere afhankelijk van de ontwerpsnelheid van de weg, geldende snelheidsbeperkingen en het weer. Op dat moment zijn de intensiteit en dichtheid nagenoeg nul.
- q_c (vtg/uur) = de maximale capaciteit van de weg
- k_c (vtg/km) = de dichtheid waarbij de capaciteit wordt bereikt
- u_c (km/uur) = de snelheid waarbij de capaciteit van de weg bereikt wordt

Het is van belang de juiste snelheid in het basisdiagram te gebruiken. Dit is namelijk de gemiddelde momentane snelheid u_M .

Momentane, harmonische en lokale gemiddelde snelheid

De gemiddelde momentane snelheid u_M is het gemiddelde van de snelheden van de voertuigen die zich op één moment op een bepaald wegvak bevinden:

$$u_m = \frac{1}{m} \sum_{i=1}^m v_i \quad [\text{B.2}]$$

waarin: u_M = gemiddelde momentane snelheid (m/s of km/uur)
 m = aantal voertuigen dat zich op één moment op een bepaald wegvak bevindt
 v_i = snelheid van voertuig i (m/s of km/uur)

De gemiddelde momentane snelheid u_M kan worden benaderd door de harmonisch gemiddelde snelheid u_H :

$$u_H = \frac{n}{\sum_{j=1}^n \frac{1}{v_j}} \quad [\text{B.3}]$$

waarin: u_H = harmonisch gemiddelde snelheid (m/s of km/h)
 n = aantal voertuigen dat binnen een bepaalde tijd een wegdoorsnede passeert;
 v_j = snelheid van voertuig j (m/s of km/uur)

De gemiddelde lokale snelheid u_L is het gemiddelde van de snelheden van de voertuigen die op één doorsnede gedurende een bepaalde periode worden gemeten:

$$u_L = \frac{1}{p} \sum_{k=1}^p v_k \quad [\text{B.4}]$$

waarin: u_L = de gemiddelde lokale snelheid (m/s of km/h)
 p = aantal voertuigen dat binnen een bepaalde tijd een wegdoorsnede passeert;
 v_k = snelheid van voertuig k (m/s of km/uur)

Bezettingsgraad

De bezettingsgraad is een grootte die staat voor de fractie van tijd dat een voertuig de detector bezet. Een voertuig dat een loop passeert “bezet” deze vanaf het moment dat de voorzijde van het voertuig aan het begin van de loop is totdat zijn achterkant aan het eind van de loop is. Voor elke voertuig bestaat een individuele bezettingsgraad periode. Voor een periode T , waarin n voertuigen passeren, kan de bezettingsgraad β gedefinieerd worden als:

$$\beta = \frac{1}{T} \sum_{i=1}^n b_i \quad [\text{B.5}]$$

waarin: b_i = individuele bezettingsgraad
 T = periode
 n = aantal gepasseerde voertuigen in periode T

Voor elke individueel voertuig is zijn bezettingsgraad afhankelijk van de snelheid die het voertuig op dat moment heeft en de lengte van dat voertuig plus de lengte van detector zelf. Voor een individueel voertuig geldt:

$$b_i = \frac{L_i + L_{loop}}{u_i} \quad [\text{B.6}]$$

waarin: L_i = lengte voertuig i
 L_{loop} = lengte detectielus
 u_i = snelheid voertuig i

Substitutie van de formules B.5 en B.6 en vermenigvuldiging met $n(1/n)$ levert het volgende:

$$\begin{aligned} \beta &= \frac{1}{T} \sum_{i=1}^n \frac{L_i + L_{loop}}{u_i} = \frac{L_{loop}}{T} \sum_{i=1}^n \frac{1}{u_i} + \frac{1}{T} \sum_{i=1}^n \frac{L_i}{u_i} \\ &= L_{loop} \cdot \frac{n}{T} \cdot \frac{1}{n} \sum_{i=1}^n \frac{1}{u_i} + \frac{1}{T} \sum_{i=1}^n \frac{L_i}{u_i} \end{aligned} \quad [\text{B.7}]$$

Indien we β substitueren formule B.2 en er vanuit gaan dat het fundamenteel diagram van toepassing ontstaat:

$$\beta = L_{loop} \cdot \frac{n}{T} \cdot \frac{1}{u_M} + \frac{1}{T} \sum_{i=1}^n \frac{L_i}{u_i} = L_{loop} \frac{q}{u_M} + \frac{1}{T} \sum_{i=1}^n \frac{L_i}{u_i} \quad [\text{B.8}]$$

Indien ervan uit kan worden gedaan dat alle voertuigen ongeveer dezelfde lengte hebben, kan β geschreven worden als:

$$\beta = L_{loop} \frac{q}{u_M} + L_i \frac{q}{u_M} = (L_{loop} + L_i) \frac{q}{u_M} = L_{tot} \frac{q}{u_M} = L_{tot} k \quad [\text{B.9}]$$

Schokgolftheorie [6]

Een schokgolf in een verkeersstroom is een grens van twee verkeerstoestanden. De snelheid van een schokgolf kan worden berekend met:

$$\omega = \frac{q_2 - q_1}{k_2 - k_1} = \frac{\Delta q}{\Delta k} \quad [B.10]$$

waarin: ω = snelheid van de schokgolf [km/uur]

Δq = verschil in intensiteiten q_1 en q_2 tussen de verkeerstoestanden 1 en 2 [vtg/uur]

Δk = verschil in dichtheden k_1 en k_2 tussen de verkeerstoestanden 1 en 2 [vtg/km]

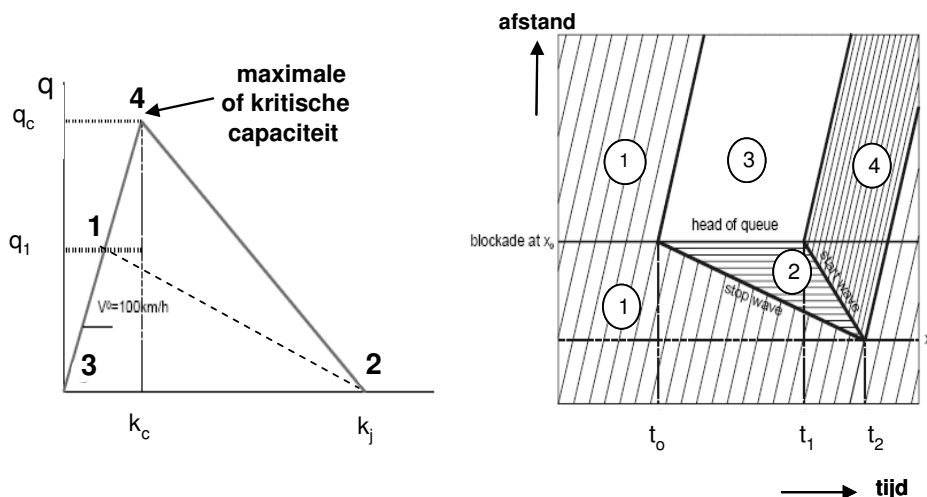
Schokgolftheorie maakt het mogelijk op basis van verkeerstoestanden de verkeersafwikkeling in plaats en tijd te bepalen. Onderstaand wordt een voorbeeld uitgewerkt waarin de verkeerstoestanden met behulp van schokgolftheorie worden besproken.

Voorbeeld schokgolftheorie

In dit voorbeeld is de weg als gevolg van een incident tussen t_0 en t_1 afgesloten. In figuur B5 is de verkeerssituatie weergegeven in het intensiteit – dichtheidsdiagram. De capaciteit van de weg is gelijk aan q_c en de stremmingsdichtheid is gelijk aan k_j .

Figuur B5:

Intensiteit-dichtheid diagram van een weg (links) en .Verkeersafwikkeling volgens schokgolftheorie indien de verkeersvraag groter is dan de restcapaciteit



In toestand 1 is sprake van vrije afwikkeling. Indien de weg geblokkeerd wordt gaat deze toestand over in toestand 2 en staan de voertuigen stil. De stopgolf heeft dan een snelheid van:

$$\omega_{stopgolf} = \frac{q_2 - q_1}{k_2 \cdot k_1} \quad [B.11]$$

waarin: q_1 = intensiteit in toestand 1
 q_2 = intensiteit in toestand 2
 k_1 = dichtheid in toestand 1
 k_2 = dichtheid in toestand 2

De kop van de file vormt een schokgolf die de overgang weergeeft tussen

toestand 2 waar het verkeer stil staat en toestand 3 waar zich geen verkeer bevindt. De snelheid waarmee deze schokgolf zich voorplant bedraagt:

$$\omega_{\text{filekop}} = 0$$

Als de blokkade voorbij is gaat toestand 2, waarbij het verkeer volledig stilstaat, over naar de maximale capaciteit q_c (toestand 4). Deze overgang wordt aangeduid met startgolf en heeft een snelheid van:

$$\omega_{\text{startgolf}} = \frac{q_c}{k_c \cdot k_j} \quad [\text{B.12}]$$

waarin: q_c = capaciteit van de weg
 k_j = stremmingsdichtheid
 k_c = kritische dichtheid

Bijlage C Matlab script detectiealgoritmes

Script Blokkadedetector

```
function [b3_incidents] = ...
    blokkade_detection(flow, speed, loc, mint, dir, a)

%% parameters voor Blokkadedetector
h_q_groot    = 0.9;    % h_q_groot
h_q_klein    = 0.05;   % h_q_klein
g_q_groot    = 0.9;    % g_q_groot
g_q_klein    = 0.05;   % g_q_klein
f_q_melden   = 0.7;    % f_q_melden
f_v_melden   = 0.5;    % f_v_melden
v_checken    = 50;     % v_checken
check_snelheid = 1;    % check_snelheid
n_up         = 2;      % aantal raaien boven
n_down       = 2;      % aantal raaien beneden
a_s          = 0.3;    % alpha stijgend
a_d          = 0.4;    % alpha dalend
cor          = 1;      % use corrected (1) or detected (0) flow values
logg        = 0;      % normal (0) or extra (1) logging for blokkade
detectie

b1_incidents=[];
b2_incidents=[];
b3_incidents=[];
minutes = length(mint);
n_locations = length(loc);
edge = max(n_up, n_down);

if minutes < 5 || n_locations < 2 + n_up + n_down
    disp(' Insufficient data for blokkade detection');
    return;
end

%% bereken de te verwachten intensiteit
exp_int = zeros(minutes, n_locations);
for j=1:n_locations
    exp_int(1, j) = flow(1, j);
end
% berken 'exponential' intensiteit
for i=2:minutes
    for j=1:n_locations
        if flow(i, j) > exp_int(i-1, j)
            exp_int(i, j) = a_s*flow(i, j) + (1-a_s)*exp_int(i-1, j);
        else
            exp_int(i, j) = a_d*flow(i, j) + (1-a_d)*exp_int(i-1, j);
        end
    end
end
end
% bereken 'corrected' intensiteit
if cor ~= 0
    cor_int = zeros(minutes, n_locations);
```

```

for i=1:minutes
    cor_int(i,1) = flow(i,1);
    cor_int(i,n_locations) = flow(i,n_locations);
    for j=2:n_locations-1
        if dir == 1
            dr = 1;
        else
            dr = -1;
        end
        qq = exp_int(i,j+dr)/max(1,exp_int(i,j));
        cor_int(i,j) = qq*flow(i,j);
    end
end
end
% bereken 'predicted' intensiteit
if dir == 1
    dr = -1;
else
    dr = 1;
end
for i=2:minutes
    for j=2:n_locations-1
        d = loc(j) - loc(j+dr);
        f = 60*d/max(1,speed(i,j));
        c = min(1,f);
        if cor == 0
            q(i,j) = c*flow(i-1,j+dr) + (1-c)*flow(i,j+dr);
        else
            q(i,j) = c*cor_int(i-1,j+dr) + (1-c)*cor_int(i,j+dr);
        end
    end
end
end
%% bereken h_q,
h_q=zeros(minutes,n_locations);
for i=2:minutes
    for j=2:n_locations-1
        h_q(i,j) = flow(i,j)/max(1,q(i,j));
    end
end
end
% bepaal richting afhankelijke parameters
if dir == 1
    dr = -1;
    down_first = 1;
    down_last = n_down;
    up_first = -n_up;
    up_last = -1;
else
    dr = 1;
    down_first = -n_down;
    down_last = -1;
    up_first = 1;
    up_last = n_up;
end
end
%% bereken n_b_matrix (als n_b =1, dan mogelijk incident)
n_b=zeros(minutes,n_locations);
for i=2:minutes-1
    for j=1+edge:n_locations-edge
        if h_q(i,j) < h_q_groot && ( h_q(i,j+dr) < h_q_groot ...
            || h_q(i+1,j) < h_q_groot || h_q(i+1,j+dr) < h_q_groot )
            n_b(i+1,j) = 1;
        end
    end
end
end

```

```

end
%% bereken v_b
v_b=zeros(minutes,n_locations);
for i=1:minutes
    for j=1+edge:n_locations-edge
        if check_snelheid == 1
            v_b(i,j) = min(speed(i,j+down_first:j+down_last));
        else
            v_b(i,j) = 1000;
        end
    end
end
end
%% bereken g_q
g_q=zeros(minutes,n_locations);
for i=3:minutes
    for j=1+edge:n_locations-edge
        q_beneden = sum(flow(i,j+down_first:j+down_last));
        mri_intensiteit_beneden = sum(exp_int(i-2,j+down_first:j+down_last));
        g_q(i,j) = q_beneden/max(1,mri_intensiteit_beneden);
    end
end
end
%% bereken f_v
f_v=zeros(minutes,n_locations);
for i=1:minutes
    for j=1+edge:n_locations-edge
        v_boven = min(speed(i,j+up_first:j+up_last));
        v_beneden = min(speed(i,j+down_first:j+down_last));
        f_v(i,j) = v_boven/max(1,v_beneden);
    end
end
end
%% bereken f_q
f_q=zeros(minutes,n_locations);
for i=4:minutes
    for j=1+edge:n_locations-edge
        q_beneden = sum(flow(i,j+down_first:j+down_last));
        mri_intensiteit_beneden = sum(exp_int(i-3,j+down_first:j+down_last));
        f_q(i,j) = q_beneden/max(1,mri_intensiteit_beneden);
    end
end
end

%% Controleer of aan de voorwaarden voor een blokkade wordt gedaan
k1 = 0;
k2 = 0;
k3 = 0;
for i=4:minutes-2
    for j=1+edge:n_locations-edge
        % indien geldt dat h_q < h_q_klein dan incident in 1 stap gedeteceerd
        % en vul matrix "b1_incidents
        if h_q(i,j) < h_q_klein
            k1=k1+1;
            b1_incidents(k1,1) = i;
            b1_incidents(k1,2) = loc(j);
            % indien geldt dat g_q < g_q_klein dan incident in 2 stappen
            gedeteceerd
            % en vul matrix "b2_incidents
            elseif g_q(i+1,j) < g_q_klein
                k2=k2+1;
                b2_incidents(k2,1) = i+1;
                b2_incidents(k2,2) = loc(j);
                % indien wordt voldaan aan alle voorwaarden voor een blokkade, dan
                % incident in 3 stappen gedeteceerd en vul matrix "b3_incidents
                elseif v_b(i+1,j) > v_checken ...
                    && n_b(i+1,j) == 1 && g_q(i+1,j) < g_q_groot ...

```

```

        && f_v(i+2,j) < f_v_melden && f_q(i+2,j) < f_q_melden
        k3=k3+1;
        b3_incidents(k3,1) = i+2;
        b3_incidents(k3,2) = loc(j);
    end
end
end

%% schrijf een rapport van alle incidentmeldinge
disp(' adding report to file "blokkade_detectie_report.txt"');
frep = fopen('./blokkade_detectie_report.txt','a');
s = [' Report blokkade detectie on ', datestr(now,0)];
fprintf(frep,'\n%s\n',s);
s = ' For more information, contact Leila van de Ven or Willem Jan Knibbe ';
fprintf(frep,'%s\n',s);
% incidenten gedetecteerd in stap 1
s = [' ', num2str(size(b1_incidents,1)), ' step 1 incidents detected:'];
fprintf(frep,'%s\n',s);
if size(b1_incidents,1) > 0
    s = ' minute location flow(t) speed(t) h_q(t) h_q(t-2)
v_beneden(t-1) g_q(t) g_q(t-1) f_v(t) f_q(t)';
    fprintf(frep,'%s\n',s);
end
for i=1:size(b1_incidents,1)
    ii = b1_incidents(i,1);
    jj = find(loc == b1_incidents(i,2));
    fprintf(frep,...
        ' %5d %10.2f %10.2f %10.2f %10.4f %10.4f %10.4f %10.4f %10.4f %10.4f
%10.4f\n', ...
        ii,b1_incidents(i,2),flow(ii,jj),speed(ii,jj), ...
        h_q(ii,jj), h_q(ii-2,jj), v_b(ii-1,jj), g_q(ii,jj), g_q(ii-1,jj),
f_v(ii,jj), f_q(ii,jj));
end
% incidenten gedetecteerd in stap 2
s = [' ', num2str(size(b2_incidents,1)), ' step 2 incidents detected:'];
fprintf(frep,'%s\n',s);
if size(b2_incidents,1) > 0
    s = ' minute location flow(t) speed(t) h_q(t) h_q(t-2)
v_beneden(t-1) g_q(t) g_q(t-1) f_v(t) f_q(t)';
    fprintf(frep,'%s\n',s);
end
for i=1:size(b2_incidents,1)
    ii = b2_incidents(i,1);
    jj = find(loc == b2_incidents(i,2));
    fprintf(frep,...
        ' %5d %10.2f %10.2f %10.2f %10.4f %10.4f %10.4f %10.4f %10.4f %10.4f
%10.4f\n', ...
        ii,b2_incidents(i,2),flow(ii,jj),speed(ii,jj), ...
        h_q(ii,jj), h_q(ii-2,jj), v_b(ii-1,jj), g_q(ii,jj),...
        g_q(ii-1,jj), f_v(ii,jj), f_q(ii,jj));
end
% incidenten gedetecteerd in stap 3
s = [' ', num2str(size(b3_incidents,1)), ' step 3 incidents detected:'];
fprintf(frep,'%s\n',s);
if size(b3_incidents,1) > 0
    s = ' minute location flow(t) speed(t) h_q(t) h_q(t-2)
v_beneden(t-1) g_q(t) g_q(t-1) f_v(t) f_q(t)';
    fprintf(frep,'%s\n',s);
end
for i=1:size(b3_incidents,1)
    ii = b3_incidents(i,1);
    jj = find(loc == b3_incidents(i,2));
    fprintf(frep,...

```

```

        ' %5d %10.2f %10.2f %10.2f %10.4f %10.4f %10.4f %10.4f %10.4f %10.4f
%10.4f\n', ...
        ii,b3_incidents(i,2),flow(ii,jj),speed(ii,jj), ...
        h_q(ii,jj), h_q(ii-2,jj), v_b(ii-1,jj), g_q(ii,jj),...
        g_q(ii-1,jj), f_v(ii,jj), f_q(ii,jj));
end
s = ' Using parameters: ';
fprintf(frep, '%s\n', s);
fprintf(frep, ' %10.2f %s\n', h_q_groot, ': h_q_groot');
fprintf(frep, ' %10.2f %s\n', h_q_klein, ': h_q_klein');
fprintf(frep, ' %10.2f %s\n', g_q_groot, ': g_q_groot');
fprintf(frep, ' %10.2f %s\n', g_q_klein, ': g_q_klein');
fprintf(frep, ' %10.2f %s\n', f_q_melden, ': f_q_melden');
fprintf(frep, ' %10.2f %s\n', f_v_melden, ': f_v_melden');
fprintf(frep, ' %10.2f %s\n', v_checken, ': v_checken');
fprintf(frep, ' %10d %s\n', check_snelheid, ': check_snelheid');
fprintf(frep, ' %10d %s\n', n_up, ': aantal raaien boven');
fprintf(frep, ' %10d %s\n', n_down, ': aantal raaien beneden');
fprintf(frep, ' %10.2f %s\n', a_s, ': alpha stijgend');
fprintf(frep, ' %10.2f %s\n', a_d, ': alpha dalend');
fclose(frep);

```

Script Presikhaaf algoritme

```

function [p3_incidents] = ...
    presik_detection(flow, speed, loc, mint, dir)

% parameters voor het Presikhaaf algoritme
par_stap1_1    = 0.6;    % parameter step 1.1
par_stap1_2    = 75;    % parameter step 1.2
par_stap1_3    = 80;    % parameter step 1.3
par_stap2_1    = 0.8;    % parameter step 2.1
par_stap2_2    = 70;    % parameter step 2.2
par_stap2_3    = 0.97;  % parameter step 2.3
par_stap3_1    = 80;    % parameter step 3.1
par_stap3_2    = 0.5;    % parameter step 3.2

p3_incidents=[];
p2_incidents=[];
p1_incidents=[];
minutes = length(mint);
n_locations = length(loc);

% Bereken het voortschrijdend gemiddelde van de snelheid en zet deze in een
% matrix "vsg_speed"
vsg_speed = zeros(minutes,n_locations);
for i=1:5
    for j=1:n_locations
        vsg_speed(i,j)=speed(i,j);
    end
end
for i=6:minutes
    for j=1:n_locations
        vsg_speed(i,j) = (speed(i-1,j)+speed(i-2,j)+speed(i-3,j)+...
            speed(i-4,j)+speed(i-5,j))/5;
    end
end

```

```

end
% Bereken het voortschrijdend gemiddelde van de intensiteit en zet deze in
een
% matrix "vsg_flow"
vsg_flow = zeros(minutes,n_locations);
for i=1:5
    for j=1:n_locations
        vsg_flow(i,j)=flow(i,j);
    end
end
for i=6:minutes
    for j=1:n_locations
        vsg_flow(i,j) = (flow(i-1,j)+flow(i-2,j)+flow(i-3,j)+...
            flow(i-4,j)+flow(i-5,j))/5;
    end
end

% Voer alle stappen van het Presikhaafalgoritme achtereenvolgens uit:

%% 1 OP TIJDSTIP t
% STAP 1.1: Als huidige snelheid veel lager dan voortschrijdendgemiddelde,
% dan mogelijk een incident
step11=zeros(minutes,n_locations);
for i=1:5
    for j=1:n_locations-1
        step11(i,j)=0;
    end
end
for i=6:minutes
    for j=1:n_locations-1
        step11(i,j) = speed(i,j)/vsg_speed(i,j)<par_stap1_1;
    end
end
% STAP 1.2: De snelheid stroomafwaarts moet hoog genoeg zijn om sprake te
% zijn van een incident (snelheidsverschil door file eruit halen)
step12=zeros(minutes,n_locations);
for i=1:minutes
    for j=2:n_locations-1
        l(i,j)=speed(i,j)/speed(i,j+dir);
        if ((l(i,j)<0.5) && (speed(i,j+dir)>par_stap1_2) &&
(speed(i,j+dir)>60))
            step12(i,j)=1;
        end
    end
end
% STAP 1.3: De actuele snelheid stroomafwaarts moet hoger zijn dan een
% drempel indien er sprake is van een incident (schokgolf eruit halen)
step13=zeros(minutes,n_locations);
for i=1:minutes
    for j=2:n_locations-1
        step13(i,j)=vsg_speed(i,j+dir)>par_stap1_3;
    end
end

%% STAP 2 (1 MINUUT LATER DUS OP t+1)
% STAP 2.1: De actuele snelheid moet laag genoeg zijn (kleiner dan
% snelheid_vsg*factor) indien er sprake is van een incident
step21=zeros(minutes,n_locations);
for i=2:minutes-1
    for j=1:n_locations-1
        l(i,j)=speed(i,j)*par_stap2_1;
        step21(i,j)=speed(i,j)<vsg_speed(i,j)*par_stap2_1;
    end
end

```

```

end
% STAP 2.2: De actuele snelheid stroomafwaarts moet groot genoeg bij een
% incident
step22=zeros(minutes,n_locations);
for i=1:minutes
    for j=2:n_locations-1
        step22(i,j)=speed(i,j+dir)>par_stap2_2;
    end
end
% STAP 2.3: Stroomafwaarts moet de intensiteit duidelijk minder zijn indien
% er sprake is van een incident
step23=zeros(minutes,n_locations);
for i=1:minutes
    for j=2:n_locations-1
        step23(i,j)=1;
        verhouding23(i,j)=flow(i,j+dir)/vsg_flow(i,j+dir);
        step23(i,j)=verhouding23(i,j)<par_stap2_3;
    end
end

%% STAP 3 (WEER 1 MINUUT LATER DUS OP t+2)
% STAP 3.1: Bij een incident moet de actuele snelheid klein genoeg zijn
step31=zeros(minutes,n_locations);
for i=1:minutes
    for j=1:n_locations-1
        step31(i,j)=speed(i,j)<par_stap3_1;
    end
end
% STAP 3.2: De actuele snelheid moet voldoende lager zijn dan de
% voortschrijdende gemiddelde snelheid
step32=zeros(minutes,n_locations);
for i=1:minutes
    for j=1:n_locations-1
        step32(i,j)=speed(i,j)<par_stap3_2*vsg_speed(i,j);
    end
end

%% Ga voor elke locatie (en elke minuut) na of aan alle voorwaarden voor een
%% incident wordt voldaan (stap 1, stap 2 en stap 3)
k1 = 0;
k2 = 0;
k3 = 0;
for i=1:minutes-2
    for j=1:n_locations
        % controleer of aan alle voorwaarden in de eerste stap wordt
        % voldaan
        if step11(i,j)==1 && step12(i,j)==1 && step13(i,j)==1;
            k1=k1+1;
            p1_incidents(k1,1) = i;
            p1_incidents(k1,2) = loc(j);
            p1_incidents(k1,3) = j;
            i1=i+1;
            % indien aan alle voorwaarden in stap 1 wordt voldaan,
            % controleer dan alle voorwaarden uit stap 2 op een minuut
            % later (i+1)
            if step21(i1,j)==1 && step22(i1,j)==1 && step23(i1,j)==1;
                k2=k2+1;
                p2_incidents(k2,1) = i1;
                p2_incidents(k2,2) = loc(j);
                i2=i+2;
                % indien aan alle voorwaarden in stap 1 en 2 wordt voldaan,
                % controleer dan alle voorwaarden uit stap 3 op weer een
                minuut
            end
        end
    end
end

```

```

        s = ' minute location flow(t) speed(t) step31 step32
speed(i,j-a) vsg_speed';
        fprintf(frep, '%s\n', s);
    end
    for i=1:size(p3_incidents,1)
        ii = p3_incidents(i,1);
        jj = find(loc == p3_incidents(i,2));
        fprintf(frep, ...
            '%5d %10.2f %10.2f %10.1f %10.1f %10.1f %10.1f\n', ...
            ii, p3_incidents(i,2), flow(ii, jj), speed(ii, jj), ...
            step31(ii, jj), step32(ii, jj), speed(ii, jj), vsg_speed(ii, jj));
    end
    s = ' Using parameters: ';
    fprintf(frep, '%s\n', s);
    fprintf(frep, ' %10.2f %s\n', par_stap1_1, ': par_stap1_1');
    fprintf(frep, ' %10.2f %s\n', par_stap1_2, ': par_stap1_2');
    fprintf(frep, ' %10.2f %s\n', par_stap1_3, ': par_stap1_3');
    fprintf(frep, ' %10.2f %s\n', par_stap2_1, ': par_stap2_1');
    fprintf(frep, ' %10.2f %s\n', par_stap2_2, ': par_stap2_2');
    fprintf(frep, ' %10.2f %s\n', par_stap2_3, ': par_stap2_3');
    fprintf(frep, ' %10.2f %s\n', par_stap3_1, ': par_stap3_1');
    fprintf(frep, ' %10.2f %s\n', par_stap3_2, ': par_stap3_2');
    fclose(frep);

```

Script Strookdetectie

```

function [sp22_incidents] = ...
    strookpercentages_detection_congestie(flow, speed, loc, mint)

% parameters strookdetectie
parflow = 545;
parflowg = 1750;
par_laag = 0.25;
par_hoog = 1.25;
parspeed = 60;

minutes = length(mint);
n_locations = length(loc);
sp22_incidents=[];
sp4cs_incidents=[];
sp4c_incidents=[];

% maak matrices aan met de rijbaanintensiteiten "total_flow" en met
% rijstrookintensiteiten "p_flow"
total_flow = sum(flow,3);
[m n lanes] = size(flow);
for i=1:minutes
    for j=1:n_locations
        for k=1:lanes
            if total_flow(i, j) > 0
                p_flow(i, j, k) = flow(i, j, k)/total_flow(i, j);
            else
                p_flow(i, j, k) = 0;
            end
        end
    end
end
flow_totaal = flow;

```

```

for k = 1:lanes
    flow_totaal(:, :, k) = sum(flow, 3);
end

% maak matrix "lanenrs" aan met het aantal rijstroken per rijbaan
load 'BPSCComplete.mat';
LaneNrs=BPSCComplete.LaneNrs;
Locations=BPSCComplete.Locations;
lanenrs=[];
for i=2:length(LaneNrs)
    if LaneNrs(i) < LaneNrs(i-1)
        lanenrs=[lanenrs LaneNrs(i-1)];
    end
end
l=length(lanenrs);
lanenrs(1, l+1)=LaneNrs(length(LaneNrs));
for i=1:length(lanenrs)
    lanenrs(2, i)=Locations(i);
end

% lees de refentietabellen voor de minimale en maximale strookaandelen voor
% congestie waar snelheid lager is dan 75 km/uur in
% (voor de uitgevoerde test is een vaste tabel aangenomen voor de
% refentiewaarden)
kl=0;
ref_matrix_str=[];
sp4cs_incidents=[];
for k=1:lanes
    if k==1
        ref_matrix_str=xlsread('A12L.xls', 'A12LL', 'B3:GU170');
    end
    if k==2
        ref_matrix_str=xlsread('A12L.xls', 'A12LL', 'B200:GU367');
    end
    if k==3
        ref_matrix_str=xlsread('A12L.xls', 'A12LL', 'B400:GU567');
    end
    if k==4
        ref_matrix_str=xlsread('A12L.xls', 'A12LL', 'B600:GU767');
    end
    j1=-1;
    j2=0;
    for j=1:n_locations
        j1=j1+2;
        j2=j2+2;
        for i=2:minutes
            kl=flow_totaal(i, j, k)/60;
            % de snelheid moet groter zijn dan -0.1 (geen meetwaarde) en
            % kleiner zijn dan 75 km/h (congestiesituatie)
            if speed(i, j, k) > -0.1
                if speed(i, j, k) < 75
                    ll=find(loc(j)==lanenrs(2, :));
                    aantal_lanes=lanenrs(1, ll);
                    if flow_totaal(i, j) > parflow*aantal_lanes
                        if flow_totaal(i, j) < parflowg*aantal_lanes
                            maxx = ref_matrix_str(kl+1, j2);
                            laag = par_laag*(ref_matrix_str(kl+1, j1));
                            hoog = par_hoog*(ref_matrix_str(kl+1, j2));
                            % controleer of de meetwaarde afwijkt van de
                            % grenswaarden voor congestie
                            if ((p_flow(i, j, k) < laag) && (speed(i-
1, j, k) > parspeed)) || ...
                                ((p_flow(i, j, k) > hoog) && (speed(i-

```

```

flowpar      = 750; % minimale intensiteit waarvoor McMaster algoritme werkt
kpar         = 0.25; % veiligheidsgrens meetwaarden
Vcritpar     = 1250; % ingestelde kritische intensiteit (qc) per rijstrook

perc_vtg_kl  = 70; % percentage voertuigen klasse klein
afm_vtg_kl   = 5;  % afmeting voertuigen klasse klein
perc_vtg_mid = 20; % percentage voertuigen klasse middel
afm_vtg_mid  = 6.5; % afmeting voertuigen klasse middel
perc_vtg_gr  = 10; % percentage voertuigen klasse groot
afm_vtg_gr   = 10; % afmeting voertuigen klasse groot
L_loop       = 1.5; % lengte meetlus

ml_incidents=[];
minutes = length(mints);
n_locations = length(loc);

%berekenen bezettingsgraad voor elke locatie en elke minuut en zet deze in
%een matrix "occ"
L_gem_vtg=( (perc_vtg_kl/100)*afm_vtg_kl) +
((perc_vtg_mid/100)*afm_vtg_mid)...
+ ((perc_vtg_gr/100)*afm_vtg_gr);
Ltot=(L_loop+L_gem_vtg)/1000;
occ=[];
for i=1:minutes
    for j=1:n_locations
        occ(i,j)=Ltot*(flow(i,j)/speed(i,j));
    end
end

% bepaal waardes van de parameters a en b in de functie
% f(occ)=b*occupancy^a met behulp van de functie 'nlinfit' (nonlinear
% least-squares regression)
for j=1:n_locations
    index=find(speed(:,j)>75);
    beta =nlinfit(occ(index,j),flow(index,j),'focc',[10000 0.9]);
    b{j} = beta;
end

% maak matrix "lanenrs" met het aantal rijstroken van de desbetreffende
% locatie
load 'BPSCComplete.mat';
LaneNrs=BPSCComplete.LaneNrs;
Locations=BPSCComplete.Locations;
lanenrs=[];
for i=2:length(LaneNrs)
    if LaneNrs(i) < LaneNrs(i-1)
        lanenrs=[lanenrs LaneNrs(i-1)];
    end
end
l=length(lanenrs);
lanenrs(1,l+1)=LaneNrs(length(LaneNrs));
for i=1:length(lanenrs)
    lanenrs(2,i)=Locations(i);
end

% ga voor elke locatie (en elke minuut) na of wordt voldaan aan de
% voorwaarden van het McMaster algoritme om te bepalen of er sprake is van
% een incident
meld=0;
for j=1:n_locations-1
    vb = b{j}(1);
    va = b{j}(2);
    jj=j+dir;

```

```

    vbjj=b{jj}(1);
    vajj=b{jj}(2);
    ll=find(loc(j)==lanenrs(2,:));
    aantal_lanes=lanenrs(1,ll);
    lljj=find(loc(j+dir)==lanenrs(2,:));
    aantal_lanesjj=lanenrs(1,lljj);
    % Vcrit is kritische rijbaanintensiteit (is de aangenomen kritische
    % intensiteit (Vcritpar) per rijstrook vermenigvuldigd met het aantal
rijstroken)
    % OCMAX is max bezettingsgraad berekend uit gocc waarbij gocc gelijk is
aan Vcrit
    Vcrit=aantal_lanes*Vcritpar;
    OCMAX= (Vcrit/(kpar*vb))^(1/va);
    Vcritjj=aantal_lanesjj*Vcritpar;
    OCMAXjj=(Vcritjj/(kpar*vbjj))^(1/vajj);
    for i=1:minutes
        if flow(i,j)>(aantal_lanes*flowpar)
            gocc = kpar * (vb * (occ(i,j)^va));
            goccjj=kpar * (vbjj * (occ(i,jj)^vajj));
            % controleer of de meetwaarde voldoet aan de voorwaarden van een
inididen:
            % op locatie x is prake van verkeerstoestand 2 (occ<ocmax &
vol<gocc(j,k))
            % of verkeerstostant 3 (occ>ocmax vol<vcrit) en stroomafwaarts is
de
            % bezettingsgraad kleiner dan ocmax (occ<ocmax)
            if ((occ(i,j)<OCMAX)      && (flow(i,j) < gocc)      &&
(occ(i,jj)<OCMAXjj)) || ...
                ((occ(i,j)>OCMAX)      && (flow(i,j) < Vcrit)  &&
(occ(i,jj)<OCMAXjj))
                % indien aan de voorwaarden van een incident wordt voldaan, vul
                % dan de matrix "meld" met alle relevante gegevens over van de
                % locatie
                meld = meld +1;
                m1_incidents(meld,1) = i;
                m1_incidents(meld,2) = loc(j);
                m1_incidents(meld,3) = speed(i,j);
                m1_incidents(meld,4) = Vcrit;
                m1_incidents(meld,5) = occ(i,j);
                m1_incidents(meld,6) = OCMAX;
                m1_incidents(meld,7) = flow(i,j);
                m1_incidents(meld,8) = gocc;
                m1_incidents(meld,9) = occ(i,jj);
                m1_incidents(meld,10) = OCMAXjj;
                m1_incidents(meld,11) = flow(i,jj);
                m1_incidents(meld,12) = goccjj;
            end
        end
    end
end
end

% schrijf een rapport van alle incidentmeldingen
disp(' adding report to file "McMaster_detection_report.txt");
frep = fopen('./McMaster_detection_report.txt','a');
s = [' Report McMaster detection on ', datestr(now,0)];
fprintf(frep,'\n%s\n',s);
s = ' For more information, contact Leila van de Ven ';
fprintf(frep,'%s\n',s);
s = [' ',num2str(size(m1_incidents,1)), ' incidents detected:'];
if size(m1_incidents,1) > 0
    fprintf(frep,'%s\n',s);
    s = '
OCMAX      flow      location      gocc      speed      Vcrit      occ
OCMAXjj+1  flowjj+1  OCMAXjj+1  goccjj+1  speedjj+1  Vcritjj+1  occjj+1
';
end

```

```
    goccj+1';
end
fprintf(frep, '%s\n', s);
for i=1:size(m1_incidents,1);
    ii = m1_incidents(i,1);
    fprintf(frep, ...
        ' %5d %12.2f %12.2f %12.0f %12.4f %12.4f %12.0f %12.1f %12.4f %12.4f
%12.0f %12.1f\n', ...
        ii, m1_incidents(i,2), m1_incidents(i,3), m1_incidents(i,4), ...
        m1_incidents(i,5), m1_incidents(i,6), m1_incidents(i,7), ...
        m1_incidents(i,8), m1_incidents(i,9), m1_incidents(i,10), ...
        m1_incidents(i,11), m1_incidents(i,12));
end
s = ' Using parameters: ';
fprintf(frep, '%s\n', s);
fprintf(frep, ' %10.2f %s\n', kpar, ': kpar');
fprintf(frep, ' %10.2f %s\n', flowpar, ': flowpar');
fprintf(frep, ' %10.2f %s\n', Vcritpar, ': Vcritpar');
fclose(frep);
```
