

MASTER

Simulatie van een met verkeerslichten beveiligd netwerk

Nijman, H.

Award date:
1972

[Link to publication](#)

Disclaimer

This document contains a student thesis (bachelor's or master's), as authored by a student at Eindhoven University of Technology. Student theses are made available in the TU/e repository upon obtaining the required degree. The grade received is not published on the document as presented in the repository. The required complexity or quality of research of student theses may vary by program, and the required minimum study period may vary in duration.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

ECB-549

1864 bse

TECHNISCHE HOGESCHOOL
EINDHOVEN
STUDIEBIBLIOTHEEK
ELEKTROTECHNIEK

SIMULATIE VAN EEN
NET VERKEERSLICHTEN
BEVEILIGD NETWERK.

H. NIJMAN.

Verslag van het afstudeerwerk uitgevoerd onder
leiding van Prof. Ir. A. Hoetman en Ir. M.P.J. Stevens,
in de periode 15 september 1971 tot 15 juni 1972.

Bij dit verslag hoort een bijlage
welke de PL/I lijsten bevat van
de behandelde programma's.

Inhoudsopgave.

I.	<u>Inleiding.</u>	2
II.	<u>Eigenschappen van het programma NETSIM.</u>	3
1.	<u>De simulatie methode.</u>	4
1.1	Inleiding.	4
1.2	De geometrie van het netwerk.	6
1.3	De geometrie van het wegvak.	8
1.4	Verkeersgedrag in het wegvak.	9
1.5	Verkeersgedrag op het kruispunt.	9
1.6	De herhalingstijd van actietijdstippen.	10
1.7	Simulatie van rijdend verkeer in een wegvak.	11
2.	<u>Verkeerstransmissie in een wegvak.</u>	13
2.1	De normale snelheidsverdeling.	13
2.2	Toepassing van de normale snelheidsverdeling.	15
2.3	De exponentiële rijtijdenverdeling.	16
2.4	Toepassing van de exp. rijtijdenverdeling bij gegeven gemiddelde- en minimum rijtijd.	20
3.	<u>De verkeersverdeling over de wegvakken.</u>	21
3.1	Methode en programma beschrijving.	21
4.	<u>Verkeersverwerking aan het einde van een wegvak.</u>	23
5.	<u>Flow-diagrammen.</u>	31
6.	<u>Data-input beschrijving.</u>	47
7.	<u>Output beschrijving van de simulatie resultaten.</u>	52
8.	<u>Toepassing.</u>	57
9.	<u>Resultaten.</u>	62
10.	<u>Aanbevelingen.</u>	63
11.	<u>Literatuur.</u>	64

I. Inleiding.

De afstudeeropdracht bestond uit het ontwerpen van een model waarmee het verkeersgedrag in een met verkeerslichten beveiligd netwerk gesimuleerd kan worden, en heeft geresulteerd in een PL/I programma NETSIM met de volgende toepassingsmogelijkheden.

- Het bestuderen van de invloed van starre verkeerslicht sequenties op verkeersstromen in een willekeurig netwerk waarbij elk gewenst verkeersaanbod gerealiseerd kan worden.
- Het onderzoeken van de kwaliteit van verkeersafhankelijke regel-algorithmen voordat deze in bestaande netwerken worden toegepast.
- Het voorspellen van verkeersparameters zoals wachttijden en filelengtes in on-line toepassingen waarbij de resultaten van verkeerstellingen aan de rand van een netwerk, direct aan het systeem worden toegevoerd.

II. Eigenschappen van het simulatie programma NETSIM.

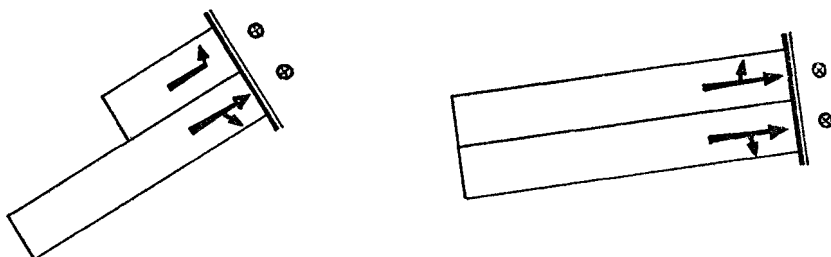
- Verkeersnetwerken van willekeurige vorm kunnen gesimuleerd worden onder de beperking dat verkeer, dat een kruispunt nadert, kan kiezen uit maximaal drie vervolgrichtingen.
- Er wordt alleen rekening gehouden met (personen) autoverkeer. Bussen, vrachtwagens e.d. moeten door middel van equivalentiecijfers omgerekend worden in personenauto's.
- Ieder verkeersaanbod kan gesimuleerd worden.
- Alle kruispunten zijn geregeerd met behulp van verkeerslichten.
- De verkeerslichten-cyclustijd is constant en geldt voor het gehele netwerk.
- De situatie in het model verandert slechts op discrete tijdstippen (actietijdstippen genaamd) welke in de orde van 2 seconden uit elkaar liggen.
- De snelheid van de voertuigen in de wegvakken is variabel en wordt gekozen uit een normale snelheidsverdeling.
- De output van het programma bestaat uit overzichten betreffende wachttijden, aantallen stops, filelengtes e.d. onderverdeeld naar de wegvakken waar deze opgetreden zijn.

1. De simulatie methode.

1.1 Inleiding.

De basiseenheid in het simulatie model is het wegvak. Hieronder wordt verstaan de rijbaan plus de opstelvakken, beschikbaar voor één rijrichting, welke twee kruispunten verbindt. Een rijbaan kan bestaan uit één of meer rijstroken.

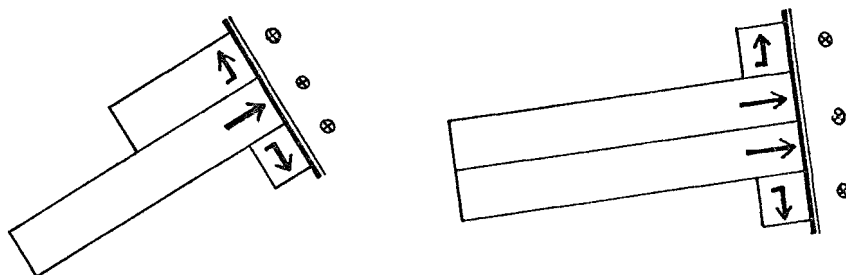
Voorbeeld.



Opstelvakken zijn die gedeelten aan de wegvakuitmonding waar verkeer met bepaalde bestemmingsrichtingen (ook wel vervolgrichtingen genaamd) zich kan opstellen. In de werkelijke situatie kunnen opstelvakken bestemd zijn voor verschillende bestemmingen; in het simulatie model bezit ieder wegvak drie opstelvakken (elk is één of meer rijstroken breed) voor de drie mogelijke vervolgrichtingen: rechtsaf, rechtdoor en linksaf.

De verkeersdoorgang wordt bestuurd door verkeerslichten, waarvan er in het model voor elke vervolgrichting één aanwezig is.

Voorbeeld.



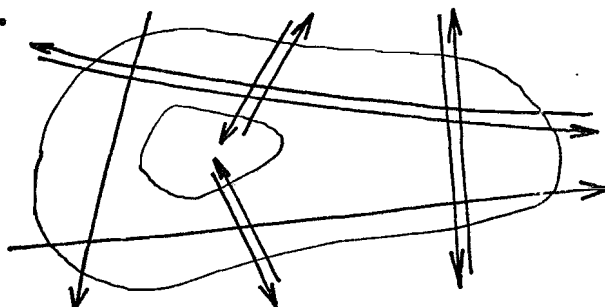
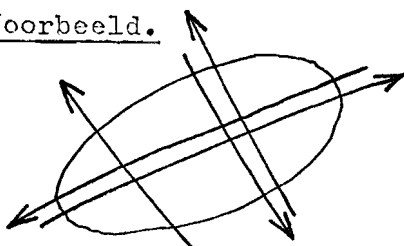
In het model stellen de voertuigen zich voor de stopstreep op, volgens in dit verslag te bespreken gedragsregels.

Een verzameling wegvakken waarvan de onderlinge relaties gegeven zijn vormen een netwerk.

In een netwerk wordt alleen autoverkeer gesimuleerd; hinder veroorzaakt door fietsers kan in rekening gebracht worden door kortere effectieve groentijden voor bepaalde verkeerslichten in te voeren.

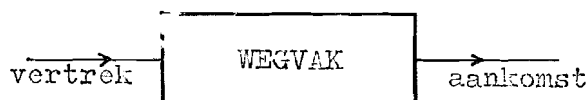
Een netwerk wordt afgesloten van de buitenwereld door één of meer randkrommen.

Voorbeeld.



Verkeer ontstaat en verdwijnt op de randkromme(n) zodat in het netwerk alleen van doorgaand verkeer sprake is.

In het simulatie model wordt onderscheid gemaakt tussen rijdende en stilstaande auto's, waarbij in het algemeen hun identiteit verloren gaat. De rijdende auto's bevinden zich per wegvak in een transmissiesysteem waarvan de input gevormd wordt door de auto's welke vanaf een kruispunt vertrekken in een bepaald wegvak.



De output van het systeem bestaat uit de aankomst van dezelfde auto's aan het einde van het wegvak. Bij constante rijtijden zal de verblijftijd in het transmissiesysteem gelijk zijn voor alle auto's.

In ons simulatie model kiest ieder voertuig zijn eigen snelheid uit een gegeven verdeling, zodat de verblijftijd in het transmissiesysteem (en daarmee de rijtijd) variabel is, en voldoet aan een bepaalde verdelingsfunctie.

De stilstaande auto's bevinden zich vóór de verschillende stopstrepen binnen het netwerk, in opstelvakken en files. Opstel- en hindermogelijkheden worden bepaald door de geometrie van de wegvakuitmondningen.

Op discrete tijdstippen welke een vaste tijd uit elkaar liggen, (actietijdstippen genaamd) wordt het gehele netwerk doorgerekend en wordt de netwerkstatus aangepast aan de veranderingen welke zich sinds het vorige actietijdstip hebben afgespeeld. Als tijdseenheid wordt 2 seconden gekozen. Voor de overgang van rijden naar stilstaan betekent dit het volgende:

Ieder actietijdstip kunnen er afhankelijk van het aantal rijstroken van het wegvak, één of meer auto's aankomen uit het transmissiesysteem. Deze worden als stilstaande voertuigen opgesteld voor de stopstreep. Indien nodig worden door middel van een loot-procedure bestemmingsrichtingen bepaald, aan de hand van de afbuigkansen.

Wanneer de betreffende verkeerslichten groen zijn, en er zich geen ander stilstaand verkeer voor de stopstreep bevindt, vertrekken de nieuw aangekomen voertuigen meteen weer en worden opgenomen in de transmissiesystemen van de bestemmingswegvakken. Optrektijden en afremtijden zijn opgenomen in de totale rijtijd van stopstreep naar stopstreep; overgang van rijden naar stilstaan is dus in het model een zeer discontinue actie.

Het hier geïntroduceerde transmissiesysteem wordt gerealiseerd door de in 1.7 te behandelen aankomstentabel.

1.2 De geometrie van het netwerk.

Een netwerk kan omsloten gedacht worden door één of meer randkrommen. Wat de verkeersstromen betreft is de netwerkgeometrie volledig vastgelegd wanneer voor alle wegvakken welke hun stopstreep binnen de randkromme(n) hebben, de bestemmingswegvakken gegeven zijn voor verkeer dat rechtsaf, recht door en linksaf gaat.

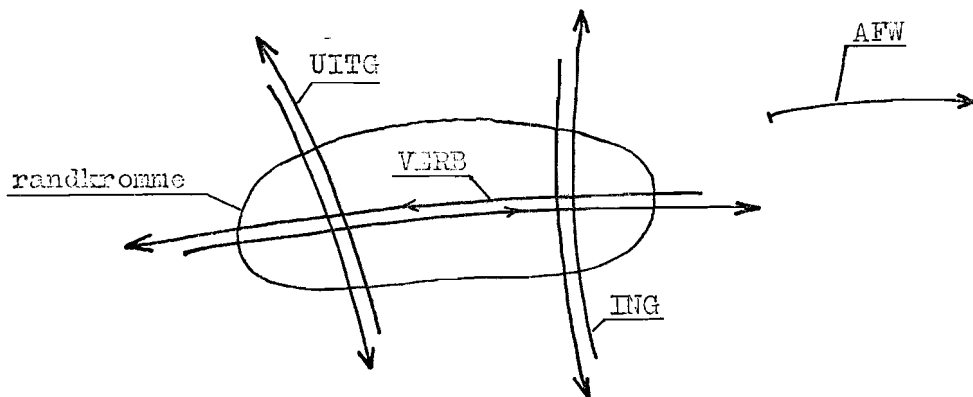
Bronnen en putten van verkeer zijn gesitueerd op de randkromme(n), namelijk op de ingaande en uitgaande wegvakken. Elk voertuig dat het netwerk binnen gaat zal na een eindige rijtijd het netwerk weer verlaten. Bij gegeven verkeers-input intensiteiten zal een verkeers-output verdeling over de uitgaande wegvakken ontstaan welke bepaald wordt door de afbuigpercentages en de verwerkingscapaciteiten van de kruispunten.

Wanneer geen blokkering optreedt zal er wat de verkeersverdeling over de wegvakken betreft, een stationnaire toestand ontstaan, waarbij voor ieder kruispunt afzonderlijk en ook voor het netwerk als geheel de Kirchhoff (verkeers)stroom vergelijking geldt. Deze stationnaire toestand wordt met een afzonderlijk programma INTEN berekend, in de vorm van de verkeersverdeling over alle wegvakken.

Wat hun relatie tot het te simuleren netwerk betreft, kunnen de wegvakken onderverdeeld worden in vier klassen. Deze klasse indeling vindt plaats aan de hand van de ligging van de vertrek- en bestemmingskruispunten.

Een wegvak behoort tot :

KLASSE = 0	wanneer het <u>afwezig</u> is, dus als beide kruispunten buiten het netwerk liggen. Ook: KLASSE = AFW.
KLASSE = 1	wanneer het <u>uitgaand</u> is, dus als het vertrekpunt wel en het bestemmingskruispunt niet binnen het netwerk ligt. Ook: KLASSE = UITG.
KLASSE = 2	wanneer het <u>ingaaand</u> is, dus als het bestemmingskruispunt wel en het vertrekpunt niet binnen het netwerk ligt. Ook: KLASSE = ING.
KLASSE = 3	wanneer het <u>verbindend</u> is, dus als beide kruispunten binnen het netwerk liggen. Ook: KLASSE = VERB.



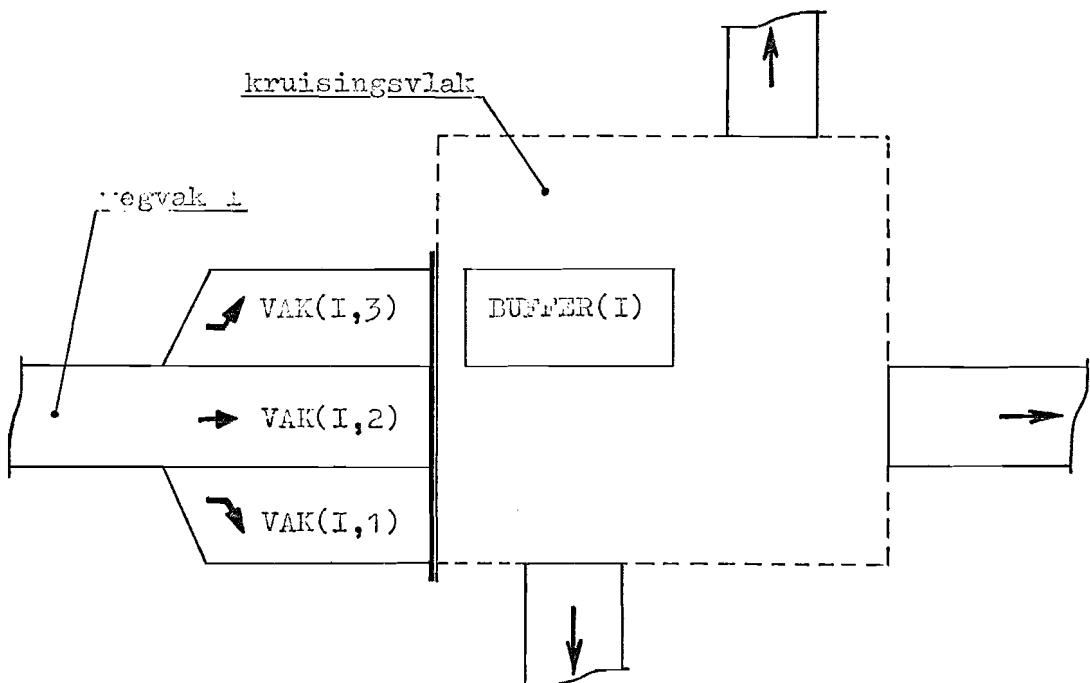
De klasse-indeling, toegelicht in bovenstaande figuur maakt een efficiënte programma- opbouw mogelijk.

Alle wegvakken waarover de simulatie zich uitstrekt, eventueel inclusief een aantal afwezige wegvakken worden beginnende bij 1, olopend genummerd. Het gevolg van het invoeren van afwezige wegvakken is dat onder handhaving van een eenmaal toegekende nummering wijzigingen van het netwerk zonder ingrijpende data-aanpassingen uitgevoerd kunnen worden.

1.3 Geometrie van het wegvak.

Karakteristiek voor een wegvak zijn:

- de lengte in meters, alleen van belang voor verbindingen wegvakken.
- de breedte van het wegvak in rijstroken. Dit komt overeen met het maximale aantal voertuigen dat per actietijdstip, naast elkaar rijdend, een willekeurig punt kan passeren. Voor veel wegvakken waarop het verkeer geconfronteerd wordt met tegenliggers geldt als feitelijke breedte 1 rijstrook, wat de mogelijkheid van inhalen echter niet uitsluit.
- de mogelijkheden om zich op te stellen voor de stopstreep. Deze hangen af van de ruimtelijke vormgeving van de wegvakuitmonding, en de toewijzing van de aanwezige ruimte (door pijlen op het wegdek) aan verkeer met bepaalde bestemmingen. Dit noodzaakt tot het onderscheiden van een aantal vormen, zoals bij de uitvoerige beschrijving van het wegvak zal blijken. (Pag. 23)
- de maximale inhoud van de buffer. Onder de buffer wordt verstaan dat gedeelte van het wegvak dat zich voorbij de stopstreep op het kruisingsvlak bevindt, waar links afslaand verkeer zich bij hinder van tegemoet komend verkeer kan opstellen. Het aantal voertuigen dat per actietijdstip de buffer kan verlaten wordt bepaald door de vormgeving van de buffer en het in te rijden wegvak.



1.4 Verkeersgedrag in het wegvak.

Naast de afhankelijkheid van de geometrie wordt het verkeersgedrag in een wegvak bepaald door:

- het opgedrukte verkeersaanbod voor de ingaande wegvakken.
- de afbuigpercentages voor afslaand verkeer.
- de rijtijdenverdeling in de verbindings wegvakken.
- de verkeerslichten instelling.

Al deze grootheden worden per wegvak constant verondersteld in de tijd.

1.5 Verkeersgedrag op het kruispunt.

Ieder wegvak met de stopstreep binnen het netwerk is in het model voorzien van drie verkeerslichten, nl. één voor iedere vervolgrichting. Wanneer een verkeerslicht groen is, en er voldoende auto's in het bijbehorende opstelvak staan, zal er per actietijdstip één voertuig uit vertrekken en geplaatst worden in het bestemmings wegvak.

Links afslaand verkeer voert deze actie in twee stappen uit; eerst wordt gereden naar de buffer, en wanneer daar geen hinder geconstateerd wordt van tegemoet komend verkeer vindt vertrek naar het bestemmings wegvak plaats.

1.6 De herhalinstijd van actietijdstippen.

Essentieel voor een betrouwbare simulatie is de grootte van de tijd tussen twee opvolgende actietijdstippen. Doordat de simulatie discreet is en de status van het netwerk dus stapsgewijs verandert, is vertrek uit een stilstaande enkelvoudige file slechts mogelijk met een frequentie van één per actietijdstip. De capaciteit van een kruispunt hangt dan ook direct af van het aantal actietijdstippen dat in de betreffende groenfasen optreedt.

Greenshields heeft een tabel gegeven^[1] voor de tijdverschillen tussen het passeren van de stopstreep van opvolgend vertrekkende voertuigen nadat het verkeerslicht groen geworden is.

volgnummer van voertuig.	verstreken tijd tussen opvolgende passages	totaal verstreken tijd sinds groenbegin
1	3.8	3.8
2	3.1	6.9
3	2.7	9.6
4	2.4	12
5	2.2	14.2
6	2.1	16.3
enz.	2.1	

De totaal verstreken tijd sinds het groenbegintijdstip is voor het n^{de} voertuig, wanneer $n \geq 5$

$$t_n = 2.1 n + 3.7 \text{ seconden.}$$

Als verschiltijd tussen twee opvolgende actietijdstippen kiezen wij 2 seconden, terwijl het groenbegintijdstip met 4 seconden vertraagd wordt. In het simulatie model geldt dan voor alle $n > 0$

$$t_n = 2n + 4 \text{ seconden.}$$

Uit een werkelijke groentijd van 24 seconden volgt een effectieve groentijd van 20 seconden, waarin dan 10 voertuigen kunnen vertrekken uit een enkelbaans wegvak.

1.7 Simulatie van rijdend verkeer in een wegvak.

In het simulatie programma van Jeuken [2] wordt het gebruik van een cyclische aankomstentabel beschreven onder toepassing van gegeven vaste rijtijden. Een aankomstentabel is een matrix waarvan de rijen zijn toegewezen aan de wegvakken, en de kolommen aan opvolgende actietijdstippen.

Wanneer tr_i de rijtijd is in wegvak i , en τ de tijd tussen twee actietijdstippen, dan volgt voor het aantal posities POS van de aankomstentabel: (het aantal kolommen)

$$POS \geq \frac{1}{\tau} \max_i [tr_i]$$

In ons simulatie model voeren we variabele rijtijden in door een snelheid of rijtijd te loten uit een gegeven verdeling. Door de mogelijkheid van optreden van extreem lange rijtijden zou het aantal posities van de aankomstentabel bijzonder groot worden. Een mogelijkheid is het aantal posities enkele malen groter te kiezen als de gemiddelde rijtijd en het langste wegvak. De weinige voertuigen met een rijtijd, groter dan POS actietijdstippen vereisen dan een bijzondere behandeling.

Een leeg wegvak wordt voorgesteld door een rij bestaande uit POS nullen. Op ieder actietijdstip is in de aankomstentabel een actuele kolom gedefinieerd. Hierin staat voor ieder wegvak het aantal auto's genoteerd dat op het bewuste actietijdstip de stopstreep zal bereiken. In ieder wegvak is het maximale aantal aankomsten per actietijdstip gelijk aan het aantal rijstroken.

Voor wat betreft het tijdstip van aankomst wordt geen rekening gehouden met de ruimtelijke aanwezigheid van eventuele files. De verwerking van de nieuw aangekomen auto's is wel afhankelijk van de aanwezigheid van reeds wachtende voertuigen.

Iedat alle voertuigen verwerkt zijn wordt de inhoud van de actuele kolom nul gemaakt.

De kolom rechts naast de actuele kolom bevat de auto's die het volgende actietijdstip zullen aankomen. Algemeen bevat de n -de kolom rechts van de actuele kolom de auto's waarvan op het actuele tijdstip bekend is dat deze over n actietijdstippen zullen aankomen, waarbij opgemerkt moet worden dat de tabel cyclisch is.

Bij vertrek vanaf een kruispunt wordt in het model voor alle voertuigen welke binnen het netwerk blijven de rijtijd tr tot de volgende stopstreep bepaald met behulp van een loot-procedure. Hiertoe moet de verdelingsfunctie van de snelheden of van de rijtijden in het betreffende wegvak bekend zijn. De auto wordt nu in de aankomstentabel geplaatst, en wel tr/τ kolommen rechts van de actuele kolom.

Na afloop van het gehele verwerkingsproces tijdens het actuele actietijdstip verschuift de actuele kolom één plaats naar rechts en begint voor het nieuwe actietijdstip de procedure opnieuw. Wanneer de actuele kolom de POS-de kolom heeft overschreden begint de actuele kolom weer vooraan in de aankomstentabel.

Loot-procedures welke nodig zijn om verkeers-input, vervolgrichtingen en snelheden (cq rijtijden) te bepalen maken gebruik van een bestaande, in assembler geschreven procedure RANDOIL. Deze levert een uniform verdeeld getal REAL, zodanig dat

$$0 < \text{REAL} < 1$$

2. Verkeerstransmissie in een wegvak.

Binnen een gesimuleerd netwerk ontstaat en verdwijnt geen verkeer. Dit betekent dat geen bronnen en putten in het netwerk liggen, en dat wanneer een voertuig zou parkeren, op hetzelfde ogenblik vanaf dezelfde plaats een ander voertuig weer aan het verkeer deelneemt. In het model van het wegvak geldt dan voor ieder voertuig een eindige rijtijd, althans ieder vertrek aan het begin van het wegvak wordt gevolgd door een aankomst op de stopstreep aan het einde ervan.

Ten aanzien van de optredende rijtijden, nodig om een wegvak af te leggen, zijn verschillende benaderingswijzen mogelijk.

Hieronder zullen besproken worden:

- de normale snelheidsverdeling, welke o.a. volgt uit spot-speed metingen en de overweging dat iedere bestuurder tot op zekere hoogte vrij is in zijn snelheidskeuze.
- de exponentiële rijtijdenverdeling, waarvan de toepassing een goede voorspelling mogelijk maakt van de cluster-ervorming.

2.1 De normale snelheidsverdeling.

In het algemeen zijn de snelheden van voertuigen, gemeten op een bepaalde plaats in een wegvak, normaal verdeeld. Men kan een gemiddelde snelheid V_0 en een spreiding s onderkennen. De waarden van V_0 en s zijn afhankelijk van de verkeersintensiteit. [3]

In ons simulatie model kunnen we nog een stap verder gaan door aan te nemen dat de gemiddelde snelheid gedurende het afleggen van een bepaald wegvak normaal verdeeld is.

Stel dat de snelheidsverdeling voldoet aan

$$f_1(v) = \frac{1}{s\sqrt{2\pi}} \exp \left[\frac{-(v-V_0)^2}{2s^2} \right]$$

waarin v de snelheid.

Deze kansverdeling kan genormeerd worden door invoering van:

gemiddelde snelheid $V_0^* = 1$

relatieve spreiding $\sigma = \frac{s}{V_0}$

snelheid $v^* = \frac{v}{V_0}$

$$f_{1N}(v^*) = \frac{1}{\sigma \sqrt{2\pi}} \exp \left[\frac{-(v^*-1)^2}{2\sigma^2} \right]$$

De benodigde rijtijd in een wegvak met de eenheid van lengte is

$$t = \frac{1}{v^*}$$

De transformatie van $f_{1N}(v^*)$ naar $f_{2N}(t)$ volgt nu met behulp van:

$$v^* = g^{-1}(t) = \frac{1}{t}$$

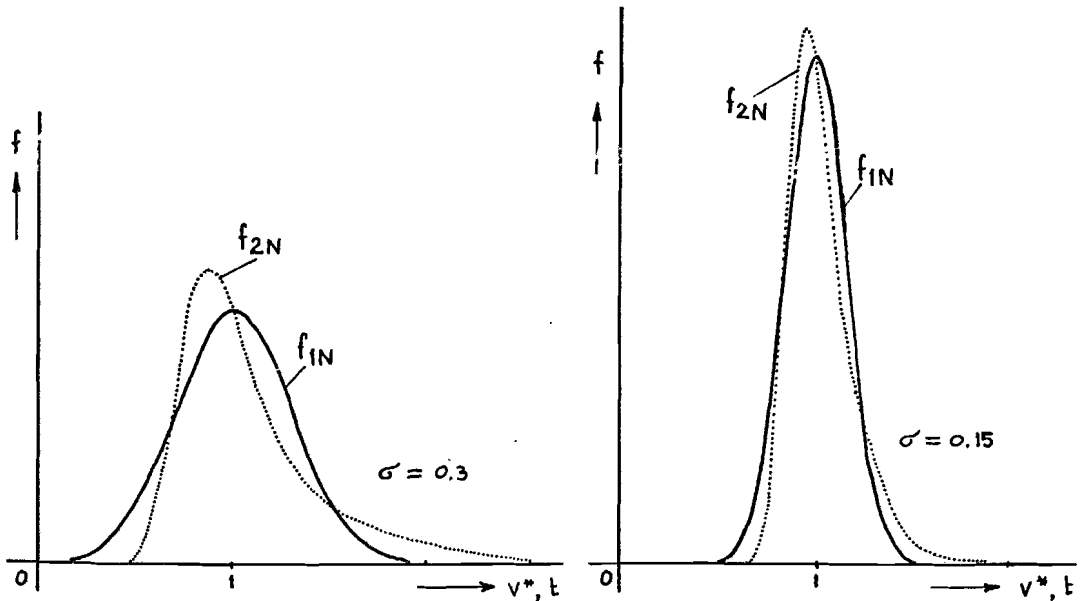
en

$$f_{2N}(t) = f_{1N} \left[g^{-1}(t) \right] \cdot \left| \frac{dg^{-1}(t)}{dt} \right|$$

zodat

$$f_{2N}(t) = \frac{1}{t^2 \sigma \sqrt{2\pi}} \exp \left[\frac{-\left(\frac{1}{t}-1\right)^2}{2\sigma^2} \right]$$

Hieronder wordt het verloop van f_{1N} en f_{2N} in beeld gebracht voor $\sigma = 0.3$ en $\sigma = 0.15$



De mediaan van de rijtijd $t_M = 1$ terwijl de gemiddelde rijtijd t_G volgt uit

$$t_G = \int_0^{\infty} t \cdot f_{2N}(t) dt$$

2.2 Toepassing van de normale snelheidsverdeling.

Een praktische moeilijkheid bij toepassing van de normale snelheidsverdeling is de bepaling van V_0 en s bij het samenstellen van de input-data. De beide grootheden kunnen niet direct gemeten worden, maar moeten per wegvak afgeleid worden uit een voldoende groot aantal metingen van de benodigde rijtijd.

In het programma NETSIM wordt op de volgende wijze een snelheid verkregen uit een normale verdeling. Eerst wordt het getal NORM uit de standaard normale verdeling ($\mu=0$, $\sigma=1$) bepaald met behulp van een aantal trekkingen van de random variabele REAL waarbij $0 < \text{REAL} < 1$; uniform verdeeld.

Volgens R.W.Hamming [4] wordt een benadering van NORM verkregen uit k trekkingen van REAL.

$$\text{NORM} = \frac{\sum_{i=1}^k \text{REAL}_i - \frac{k}{2}}{\sqrt{\frac{k}{12}}}$$

De overeenkomst van de NORM waarden met de standaard normale verdeling neemt asymptotisch toe voor toenemende k .

In ons toepassingsgebied kunnen we genoegen nemen met een vrij grove benadering, en kiezen $k=12$.

$$\text{NORM} = -6 + \sum_{i=1}^{12} \text{REAL}_i$$

De te bepalen snelheid v wordt nu

$$v = V_0 + \text{NORM} \times s$$

2.3 De exponentiële rijtijdenverdeling.

In het programma TRANSYT [5] maakt D.I. Robertson gebruik van een exponentiële rijtijdenverdeling, en doet daarmee goede voorspellingen van de clustervervorming die over een bepaald traject optreedt. De aankomst-kansverdeling wordt uit de voertuigdichtheid van een vertrekkende cluster afgeleid met behulp van de recurrenente betrekking

$$q'_{T_m+i} = F \cdot q_i + (1-F) \cdot q'_{T_m+i-1} \quad (1)$$

q_i is de voertuigdichtheid in het i -de tijdinterval van de cluster aan het begin van het wegvak.

q'_i is de voertuigdichtheid in het i -de tijdinterval zoals deze voorspeld wordt aan het einde van het wegvak.

T_m is de minimaal mogelijke rijtijd.

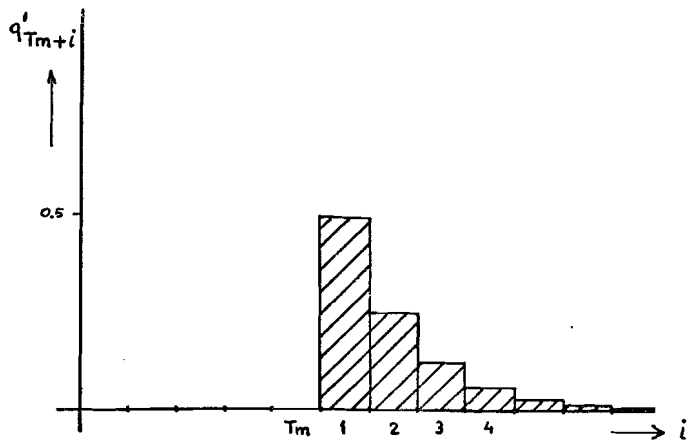
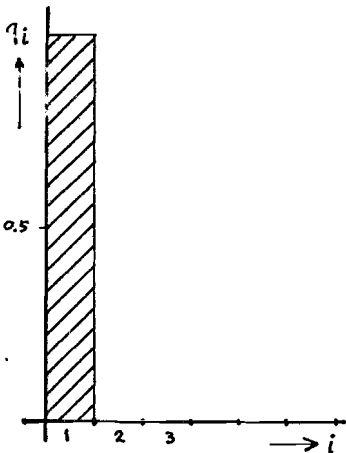
F is een afvlakfactor. Er moet gelden $0 < F \leq 1$
Robertson stelt:

$$F = \frac{1}{1+0.5T_m}$$

Passen we bovenstaande betrekking toe op één vertrekkend voertuig:

$$\begin{cases} q_i = 1 & \text{voor } i = 1 \\ q_i = 0 & \text{voor } i \neq 1 \end{cases}$$

dan vormt q'_{T_m+i} een meetkundige reeks voor $i = 1, 2, 3, \dots$



$q'_{T_{m+i}}$ is de kans op aankomst in een tijdsinterval t_2-t_1

$$\begin{aligned} \text{met } t_1 &= T_{m+i-1} \\ t_2 &= T_{m+i} \end{aligned}$$

Er geldt:

$$q'_{T_{m+i}} = (1-F) \cdot q'_{T_{m+i-1}} \tag{Ia}$$

$$\text{en } \lim_{N \rightarrow \infty} \sum_{i=1}^N q'_{T_{m+i}} = F \cdot \lim_{N \rightarrow \infty} \sum_{i=0}^N (1-F)^i = 1 \tag{Ib}$$

$$\text{met } 0 < F \leq 1$$

Definieren we de rijtijdenverdeling zo dat de kans op aankomst in hetzelfde interval is:

$$q'_{T_{m+i}} = \int_{t_1}^{t_2} \lambda e^{-\lambda(t-T_m)} dt$$

dan volgt

$$q'_{T_{m+i}} = (1-e^{-\lambda}) \cdot e^{-\lambda(i-1)}$$

$$q'_{T_{m+i}} = e^{-\lambda} \cdot q'_{T_{m+i-1}} \tag{IIa}$$

$$\text{en } \lim_{N \rightarrow \infty} \sum_{i=1}^N q'_{T_{m+i}} = (1-e^{-\lambda}) \cdot \lim_{N \rightarrow \infty} \sum_{i=0}^N e^{-\lambda \cdot i} = 1 \tag{IIb}$$

Vergelijken we (Ia) met (IIa), en (Ib) met (IIb) dan zijn beide beschrijvingswijzen identiek wanneer gekozen wordt:

$$e^{-\lambda} = 1-F \quad \text{of} \quad \lambda = \ln \frac{1}{1-F}$$

Voor $F \ll 1$ nadert λ tot F zoals volgt uit de reeksontwikkeling van

$$-\lambda = \ln(1+(-F)) \quad \text{met } 0 < F \leq 1$$

$$\lambda = F \cdot \left(1 + \frac{F}{2} + \frac{F^2}{3} + \frac{F^3}{4} + \frac{F^4}{5} + \dots \right)$$

Uit de gedefinieerde continue rijtijdenverdeling verkrijgen we als verdelingsfunctie van \underline{t}

$$Q'(t) = P(\underline{t} \leq t) = \int_{T_m}^t \lambda e^{-\lambda(\tau - T_m)} d\tau = 1 - e^{-\lambda(t - T_m)} \quad \text{met } t \geq T_m$$

en als verdelingsdichtheid van \underline{t}

$$f(t) = \frac{d}{dt} Q'(t) = \lambda e^{-\lambda(t - T_m)}$$

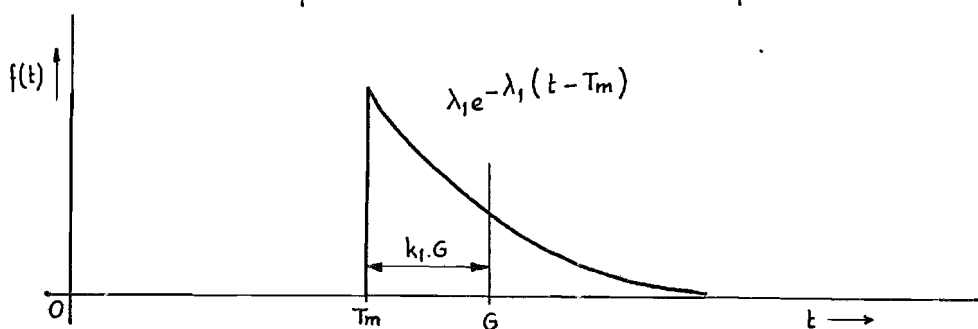
De door Robertson toegepaste betrekking (1) is een discretisatie van bovenstaande continue verdeling waarbij

$$F = 1 - e^{-\lambda}$$

Geval 1.

Zij gegeven: de rijtijdenverdeling is exponentieel met gemiddelde rijtijd G en minimale rijtijd T_m .

Definieer k_1 zodanig dat $(G - T_m) = k_1 \cdot G$



$$f(t) = \lambda_1 \cdot e^{-\lambda_1(t - T_m)} \quad \text{voor } t \geq T_m$$

$$f(t) = 0 \quad \text{voor } t < T_m$$

Opdat G het gemiddelde is geldt:

$$\int_{T_m}^{\infty} t \cdot f(t) dt = G$$

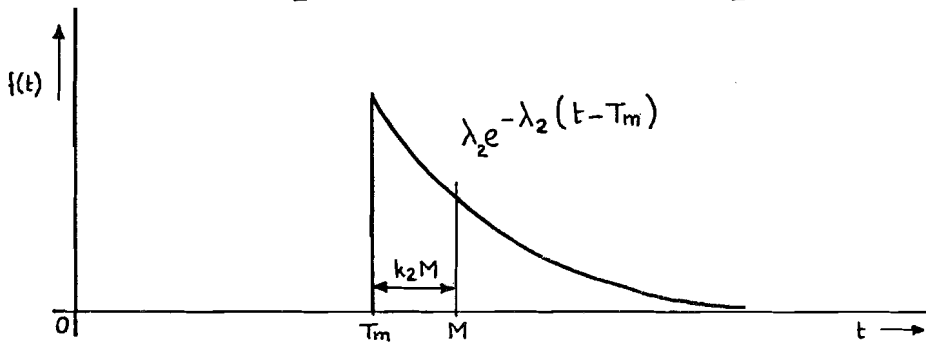
$$\lambda_1 \int_{T_m}^{\infty} t \cdot e^{-\lambda_1(t-T_m)} dt = \frac{\lambda_1 T_m + 1}{\lambda_1} = G$$

$$\lambda_1 = \frac{1}{k_1 \cdot G}$$

Geval 2.

Zij gegeven: de rijtijdenverdeling is exponentieel met mediaan M en minimale rijtijd T_m .

Definieer k_2 zodanig dat $(M-T_m) = k_2 \cdot M$



$$f(t) = \lambda_2 \cdot e^{-\lambda_2(t-T_m)} \quad \text{voor } t \geq T_m$$

$$f(t) = 0 \quad \text{voor } t < T_m$$

Opdat M de mediaan is geldt:

$$\int_{T_m}^M f(t) dt = \frac{1}{2}$$

$$\lambda_2 \int_{T_m}^M e^{-\lambda_2(t-T_m)} dt = e^{-\lambda_2(M-T_m)} = e^{-\lambda_2 k_2 M} = \frac{1}{2}$$

$$\lambda_2 = \frac{\ln 2}{k_2 M}$$

De waarde van F uit (1) wordt nu verkregen:

in geval 1 bij gegeven
 gemiddelde- en minimum rijtijd $F = 1 - e^{-\frac{1}{k_1 G}}$

in geval 2 bij gegeven
 mediaan- en minimum rijtijd $F = 1 - \frac{1}{2}e^{-\frac{1}{k_2 M}}$

Wanneer $F = \frac{1}{1+kG}$ geldt altijd $0 < F \leq 1$

terwijl voor grote kG de waarde van F nadert tot $1 - e^{-\frac{1}{kG}}$

De door Robertson voorgestelde $F = \frac{1}{1+0.5T_m} = \frac{1}{1+0.5(1-k)G}$
 mag dus alleen als benadering
 toegepast worden als $k \approx 1/3$

2.4

Toepassing van de exponentiële rijtijdenverdeling
 bij gegeven gemiddelde en minimum rijtijd.

Stel $k = \frac{G-T_m}{G}$ dan volgt $\lambda = \frac{1}{kG}$

$$f(t) = e^{-\lambda(t-T_m)}$$

$$P(\underline{t} \leq t) = 1 - e^{-\lambda(t-T_m)} \quad 0 \leq P(\underline{t} \leq t) \leq 1$$

Trek een random getal R met $0 < R < 1$

Stel $R = P(\underline{t} \leq t)$

$$R = 1 - e^{-\lambda(t-T_m)}$$

$$1-R = r = e^{-\lambda(t-T_m)} \quad \text{met } 1 > r > 0$$

$$-\lambda(t-T_m) = \ln(r)$$

$$t = T_m - \frac{1}{\lambda} \ln(r)$$

$$= T_m - kG \ln(r)$$

Op eenvoudige wijze wordt dus een rijtijd t getrokken uit een gegeven exponentiële verdeling, daar voor r het random getal REAL genomen kan worden.

Hoewel in het simulatie programma geen gebruik wordt gemaakt van de exponentiële rijtijdenverdeling, kan hiertoe gemakkelijk worden overgegaan wanneer invoering op grond van praktijk-metingen gewenst is. Als voordeel boven de normale snelheidsverdeling geldt dan de snellere methode om een rijtijd te bepalen.

3. De verkeersverdeling over de wegvakken.

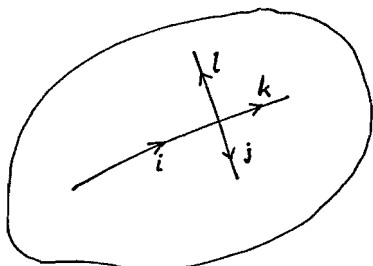
Bij gegeven netwerkgeometrie, verkeers-input intensiteiten en afbuigpercentages in alle wegvakken, kan uitgaande van de Kirchhoff stroomvergelijking de stationnaire verkeersverdeling berekend worden. Hierbij wordt geen rekening gehouden met capaciteiten, hinder en blokkering.

Het hiertoe geschreven PL/I programma INTEN werkt op dezelfde data-set als het simulatieprogramma NETSIM. Om deze reden kan met behulp van INTEN een beperkte data-input controle uitgevoerd worden, door de berekende verkeersverdelingen te vergelijken met de meetgegevens uit de praktijk, waaruit juist de data-set is samengesteld.

3.1 Methode en programmabeschrijving

Uitgangspunt is een leeg netwerk. Voor de ingaande wegvakken is de inkomende verkeers intensiteit gegeven in voertuigen per uur. In het programma wordt gerekend met de intensiteit per seconde.

In een netwerk wordt onder de rondgaande versterking van een maas verstaan: het product van de kansen (op ieder te passeren kruispunt) dat het verkeer de maas voltooid. Er wordt alleen dan een stationnaire bezetting van het netwerk gevonden wanneer er geen mazen voorkomen met rondgaande versterking gelijk aan 1. Dit zouden dan namelijk verkeersputten zijn waarvan de inhoud onbeperkt toeneemt.



Voor alle wegvakken van het netwerk worden de heersende intensiteiten verdeeld over de betreffende bestemmingswegvakken. Zo zal de verkeersstroom i zich splitsen in de drie stromen j , k en l . Op deze wijze zal iedere verkeersstroom zich blijven verdelen over de verbinding- en uitgaande wegvakken, totdat de som van de

intensiteiten in de uitgaande wegvakken gelijk wordt aan de intensiteit van het binnenkomende verkeer.

In het programma wordt gerekend met de array's:

INT bevat van alle wegvakken de nog te verdelen verkeersintensiteiten.

INTENS hierin wordt per wegvak de eindtoestand opgebouwd.

BESTEMMING bevat voor alle wegvakken de bestemmingswegvakken bij keuze rechtsaf, rechtdoor of linksaf.

RICH bevat voor alle bestemmingswegvakken de kans op inrijden.

In het programma wordt begonnen met het verdelen van de ingaande verkeersstromen INT over de betreffende bestemmingswegvakken aan de hand van de afbuigkansen RICH. De dan verkregen intensiteiten worden daarna zo lang verdeeld en gesommeerd in INTENS tot in alle wegvakken de nog te verdelen intensiteiten INT kleiner zijn dan 0,00001 voertuigen per seconde. In feite bestaat het programma uit één DO-loop welke zo vaak doorlopen wordt, totdat de stationnaire toestand bereikt is.
In PL/I:

```
AO: DO I=1 TO AANTAL_WEGVAKKEN;
      IF (KLASSE(I)=INGAAND | KLASSE(I)=VERBINDEND)&INT(I) > .00001
      THEN DO; INTENS(I)=INTENS(I)+INT(I);
              DO J=1,2,3; BES=BESTEMMING(I,J);
                  IF BES > 0
                  THEN INT(BES)=INT(BES)+INT(I)*RICH(I,J);
              END; INT(I)=0;
      END;
END;
DO I=1 TO AANTAL_WEGVAKKEN;
      IF INT(I) > .00001 THEN GOTO AO;
END;
```

Het programma INTEN levert in tabelvorm voor alle wegvakken de intensiteiten per uur, per verkeerslichtencyclustijd en per simulatierun. Dit laatste maakt het mogelijk direct een vergelijking met de simulatie resultaten te maken.

4. Verkeersverwerking aan het einde van een wegvak.

Het is duidelijk dat het verkeersgedrag voor de stopstreep in belangrijke mate bepaald wordt door de ruimtelijke vormgeving van de wegvak uitmonding. Om deze reden wordt aan elk wegvak een VORM toegekend, afhankelijk van de werkelijke geometrie.

VORM = 5. Het wegvak heeft drie of meer rijstroken en/of het voorsorteren voor de drie mogelijke vervolgrichtingen en het vullen van de opstelvakken kan uitgevoerd worden zonder onderlinge hinder. Opstelvakken zijn gedeelten van het wegvak, vlak voor de stopstreep, toegewezen aan één of meer vervolgrichtingen, van waaruit voertuigen het kruispunt kunnen oprijden.

VORM = 4. Het wegvak heeft één of twee rijstroken, en voor de stopstreep kunnen zich twee files naast elkaar opstellen met bestemming rechtdoor.

(VORM < 4.) Het wegvak heeft één rijstrook tot voor de stopstreep met de mogelijke aanwezigheid van opstelvakken voor rechtsaf- en linksafslaand verkeer. We spreken van:

VORM = 3. Het linker opstelvak is groter dan het rechter.

VORM = 2. Beide opstelvakken zijn even groot of afwezig.

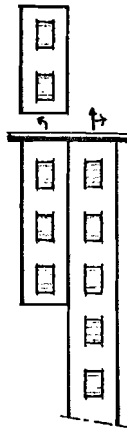
VORM = 1. Het rechter opstelvak is groter dan het linker.

In het model worden alle wegvakken welke een kruispunt naderen, voorzien van drie opstelvakken waarbij hun maximale capaciteit (maximum aantal te herbergen voertuigen) afgeleid wordt uit de afmetingen van de eventueel aanwezige opstelvakken. Deze herleiding vindt plaats in het data-invoer gedeelte van het programma.

In het nu volgende voorbeeld zal uitgaande van een werkelijke situatie het bijbehorende model opgebouwd worden. Hierbij zullen de nodige variabelen geïntroduceerd en de verwerking van het verkeer behandeld worden.

Voorbeeld.

a) De werkelijke situatie.



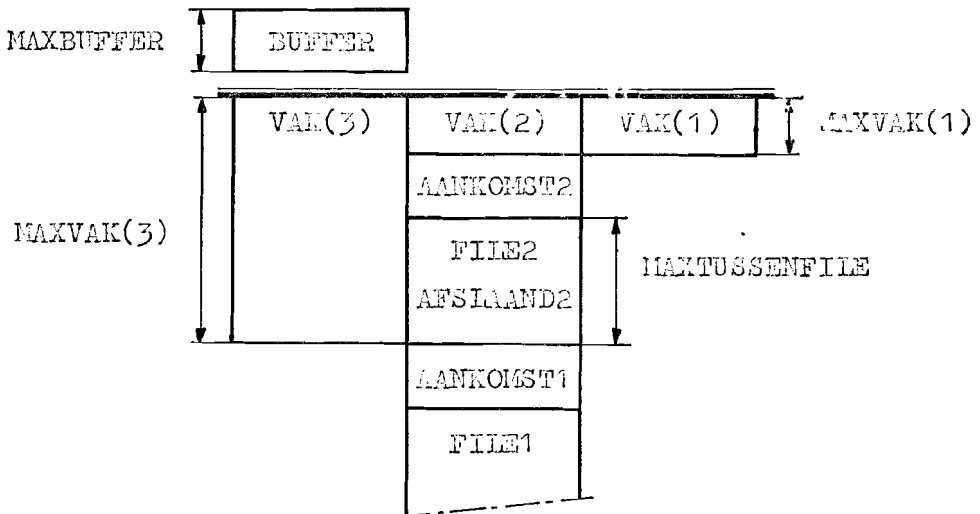
Het wegvak heeft één rijstrook welke zich voortzet tot voor de stopstreep.

Er is een opstelvak voor links afslaand verkeer met maximale inhoud MAXL auto's. In de buffer kunnen MAXBUFFER auto's.

Als data-input wordt opgegeven:

MAXL = 3;
MAXR = 0;
MAXBUFFER = 2;

b) Het simulatie model.



De opstelvakken welke ingevoerd zijn voor rechtsaf, rechtdoor en linksaf heten resp. VAK(1), VAK(2) en VAK(3), en hebben elk een uitgangs breedte van één auto. Hun maximum inhouden volgen dit:

MAXVAK(1) = MAXR + 1 = 1
MAXVAK(3) = MAXL + 1 = 4
MAXVAK(2) = min(MAXR, MAXL) + 1 = 1
MAXTUSSENFILE = abs(MAXR-MAXL) - 1 = 2

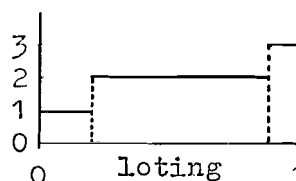
Verder zijn ingevoerd de aankomstvakken AANKOMST1 en AANKOMST2 ter lengte van één auto, waartussen ruimte is voor FILE2. Hiervan is de maximale lengte MAXTUSSENFILE voertuigen.

AFSLAAND2 geeft aan hoeveel auto's van FILE2 willen afslaan, in dit voorbeeld rechtsaf.

AANKOMST1 kan gezien worden als de toegangspoort tot VAK(3) of FILE2, en AANKOMST2 als de toegangspoort tot VAK(2) of VAK(1).

Een auto uit FILE1 krijgt door middel van een loot-procedure een bestemmingsrichting toegewezen in de vorm van een getal:

1.... rechtsaf
2.... rechtdoor
3.... linksaf



Dit getal wordt geplaatst in AANKOMST1 en de inhoud van FILE1 neemt af met 1. Om het vak AANKOMST1 weer te kunnen verlaten mag de doorgang niet geblokkeerd zijn, zodat moet gelden:

$$VAK(3) < MAXVAK(3) \wedge FILE2 < MAXTUSSENFILE$$

Is doorgang mogelijk dan neemt VAK(3) of FILE2 toe met 1 en wordt AANKOMST1 weer 0. In het geval dat de bestemming rechtsaf was wordt tevens AFSLAAND2 met 1 opgehoogd.

Een auto uit FILE2 krijgt vervolgens een bestemmingsrichting toegewezen met kans van $(AFSLAAND2 / FILE2)$ op rechtsaf. Het resultaat (1 of 2) wordt in AANKOMST2 geplaatst en de inhoud van FILE2 en AFSLAAND2 wordt aangepast.

Om het vak AANKOMST2 te kunnen verlaten moet gelden:

$$VAK(2) < MAXVAK(2) \wedge VAK(1) < MAXVAK(1)$$

Wanneer doorgang mogelijk is, (in dit voorbeeld zijn dan zowel VAK(1) als VAK(2) leeg) wordt één van beide opstelvakken gevuld.

Bij de verwerking van FILE2 wordt geen rekening gehouden met de volgorde van de voertuigen wat hun bestemmingen betreft, want bij het verlaten van FILE2 wordt een vervolgrichting geloot, terwijl in feite de vervolgrichting al bekend was bij het komen in FILE2.

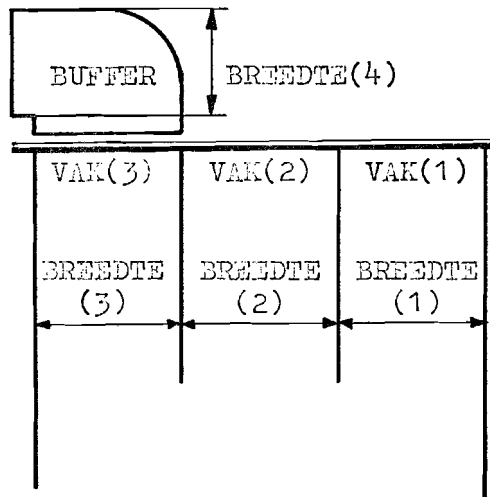
Met nadruk wordt er op gewezen dat de inhoud van alle ingevoerde vakken gegeven wordt in aantallen voertuigen, met uitzondering van AANKOMST1 en AANKOMST2 waarin een getal de vervolgrichting van het betreffende voertuig aangeeft.

AANKOMST1 = 0 betekent dat het vak leeg is.

In het model is aan elk opstelvak VAK, een verkeerslicht toegevoegd. Deze krijgen de namen LICHT(1), LICHT(2) en LICHT(3), uiteraard resp. voor rechtsaf, rechtdoor en linksaf. Een LICHT kan rood of groen zijn.

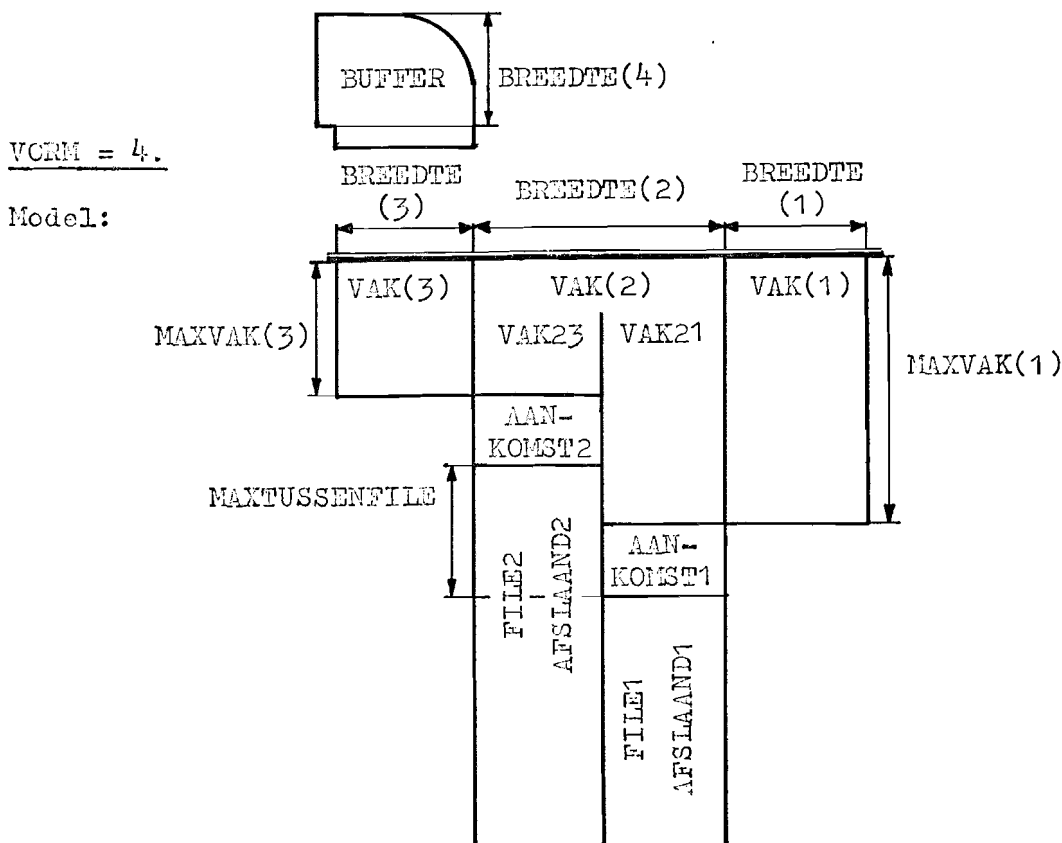
VORM = 5.

Model:



De uitgangsbreedte van de opstelvakken en de buffer (het aantal voertuigen dat naast elkaar rijdend gelijktijdig kan vertrekken) wordt gegeven door $BREEDTE(i)$ met $i \in \{1,2,3,4\}$

Aangenomen wordt dat alle reeds aangekomen voertuigen zijn opgenomen in VAK(1), VAK(2) en VAK(3). Elk volgend actietijdstip zullen de dan aankomende voertuigen zich zonder onderlinge hinder opstellen in de gewenste opstelvakken. Deze zijn niet aan een maximale inhoud gebonden. Ook kunnen voertuigen zich niet zo opstellen dat een gedeelte van het wegvak geblokkeerd wordt.



Gegeven zijn:

- MAXR de maximale inhoud van het opstelvak voor rechtsaf.
- MAXL de maximale inhoud van het opstelvak voor linksaf.
- BREEDTE(4) de uitgangsbreedte van de buffer.

Hieruit worden berekend: $MAXVAK(1) = MAXR + 1$
 $MAXVAK(3) = MAXL + 1$
 $MAXTUSSENFILE = MAXR - MAXL$

dus $MAXTUSSENFILE < 0$ als $MAXR < MAXL$.

Verder zijn bekend:

$$BREEDTE(1) - BREEDTE(3) = 1$$

$$BREEDTE(2) = 2$$

Voertuigen die eenmaal opgenomen zijn in FILE1 of FILE2 overschrijden de scheidelingslijn tussen linker en rechter rijstrook niet meer.

De rechter rijstrook is niet geblokkeerd als:

$$VAK(1) < MAXVAK(1) \wedge VAK21 < MAXVAK(1)$$

Hetzelfde geldt voor de linker rijstrook als:

$$VAK(3) < MAXVAK(3) \wedge VAK23 < MAXVAK(3)$$

De verkeersverwerking verloopt als volgt:
Tijdens het actuele actietijdstip zullen afhankelijk van de inhoud van de aankomstentabel n voertuigen de stopstreep bereiken.

$$n \in \{0, 1, 2\}$$

n=2: Door middel van een richting bepalende hulp-procedure wordt voor beide auto's een vervolgrichting RICHT1 en RICHT2 geloot. Doordat voor afbuigend verkeer de rijstrook keuze vast ligt, en in het programma de twee auto's na elkaar verwerkt worden, is het van voordeel RICHT1 en RICHT2 te verwisselen wanneer RICHT1 = 2. (rechtdoor)
Als resultaat verkrijgen we de volgende combinaties.

RICHT1	1	1	1	1	2	3	3	3	3
RICHT2	1	2	3	2	2	2	1	2	3

Door eerst het voertuig met RICHT1 te verwerken wordt de rijstrook keuze:

R	R	R	R	R	L	L	L	L
R	L	L	L	L	R	R	R	L

Afhankelijk van de rijstrook keuze wordt FILE1 of FILE2 opgehoogd evenals eventueel AFSLAAND1 en AFSLAAND2. De combinaties RR en LL worden hier wel toegestaan, hoewel dit strijdig is met het "naast elkaar rijdend gelijktijdig de stopstreep bereiken." Dit om zo goed mogelijk aan de afbuigpercentages, en daarmee aan de gewenste verkeersstroming te voldoen.

n=1. Na richting bepaling volgt:

RICHT1	1	2	3
RICHT2	0	0	0

met bijbehorende rijstrook keuze:

R	?	L
---	---	---

? is de rijstrook met de kortste file gemeten vanaf de stopstreep. FILE1, FILE2, AFSLAAND1 of AFSLAAND2 worden indien nodig opgehoogd.

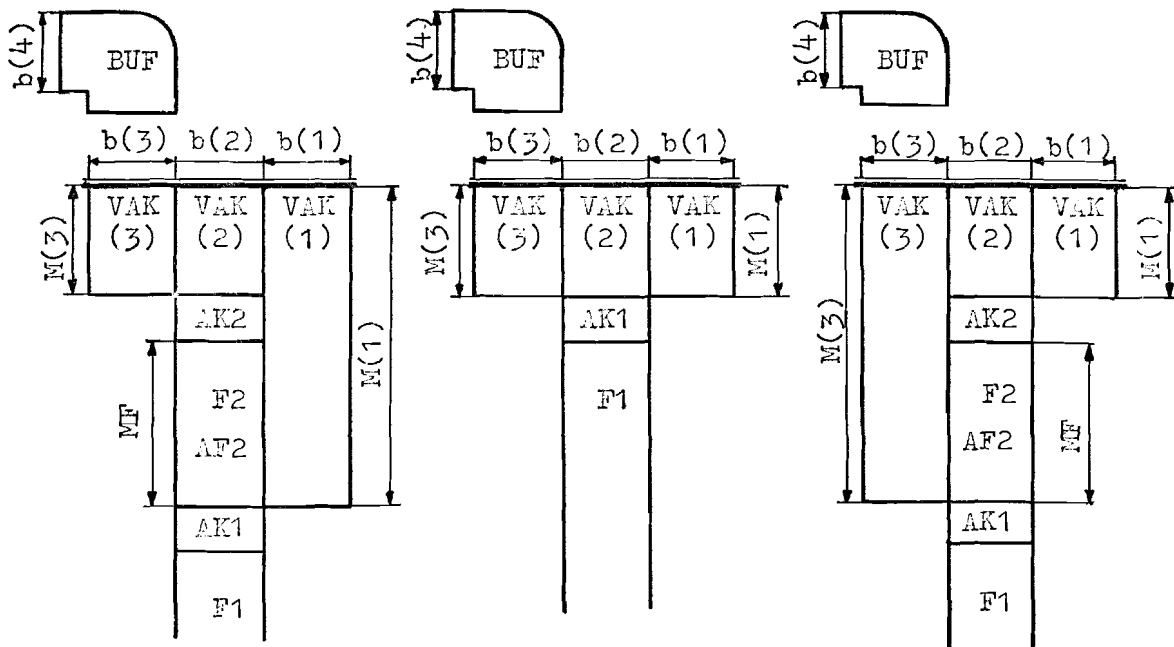
n=0. Geen nieuwe voertuigen komen aan.

Voor alle n wordt hierna bij aanwezigheid van file(s) na een hernieuwde loting t.o.v. de fractie (AFSLAAND / FILE) het bijbehorende vak AANKOMST gevuld met een bestemmings getal. Zonder blokkade kan de auto doorrijden van AANKOMST1 naar VAK(1) of VAK21; cq. van AANKOMST2 naar VAK(3) of VAK23.

VORM = 1.

VORM = 2.

VORM = 3.



F1 = FILE1
 F2 = FILE2
 AK1 = AANKOMST1
 AK2 = AANKOMST2
 BUF = BUFFER

AF2 = AFSLAAND2
 M(i) = MAXVAK(i)
 MF = MAXTUSSENFILE
 b(4) = BREEDTE(4) (uitgangsbreedte buffer)

Als gegevens, afgeleid uit de werkelijke situatie worden ingevoerd:

MAXR	}	VORM = 1	als	MAXR > MAXL
MAXL		VORM = 2	als	MAXR = MAXL
BREEDTE(4)		VORM = 3	als	MAXR < MAXL

Hieruit worden berekend:

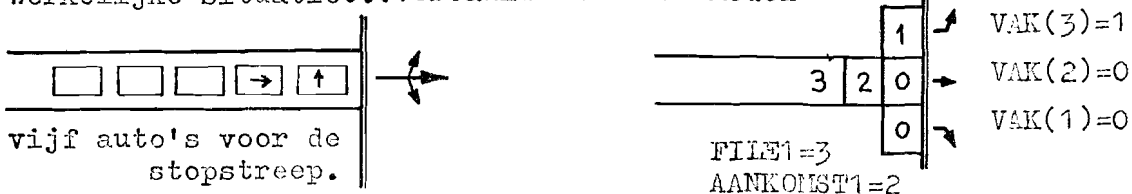
MAXVAK(1) = MAXR + 1
 MAXVAK(3) = MAXL + 1
 MAXVAK(2) = min(MAXR, MAXL) + 1
 MAXTUSSENFILE = abs(MAXR - MAXL) - 1

Verder geldt: BREEDTE(1) = BREEDTE(2) = BREEDTE(3) = 1

Wanneer VORM = 2 hebben FILE2, AFSLAAND2 en AANKOMST2 geen betekenis.

Voorbeeld. MAXR = MAXL = 0

Werkelijke situatie...vertaald naar het model



Het rijgedrag in het model wanneer $VORM \leq 3$ is als volgt:
Een nieuw aangekomen voertuig wordt opgenomen in FILE1.
Wanneer AANKOMST1 = 0 wordt een vervolgrichting geloot, en
AANKOMST1 gevuld met het resultaat.

Als $VORM = 2$ gaat de auto door naar $VAK(AANKOMST1)$ mits:

$$VAK(1) < MAXVAK(1) \wedge VAK(2) < MAXVAK(2) \wedge VAK(3) < MAXVAK(3)$$

De verwerking in het geval $VORM = 2$ is hiermee beëindigd.

Wanneer $AANKOMST1 = VORM$ gaat het voertuig naar $VAK(VORM)$
mits:

$$VAK(VORM) < MAXVAK(VORM) \wedge FILE2 < MAXTUSSENFILE$$

Wanneer $AANKOMST1 \neq VORM$ gaat de auto uit AANKOMST1
naar FILE2 en indien AANKOMST2 leeg is gaat er ook een
auto van FILE2 naar AANKOMST2. We nemen hiervoor dezelfde
bestemmingsrichting als van de auto die achter aansloot
bij FILE2. Zou echter AANKOMST2 al gevuld zijn, dan wordt
FILE2 en eventueel AFSLAAND2 opgehoogd met 1.

Van AANKOMST2 gaat het voertuig naar $VAK(AANKOMST2)$ mits:

$$VAK(2) < MAXVAK(2) \wedge VAK(4-VORM) < MAXVAK(4-VORM)$$

-+-+-+-----

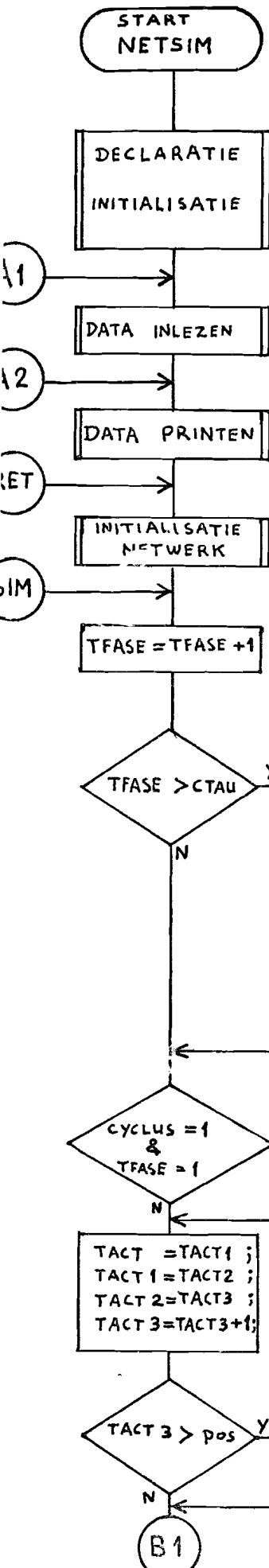
Het rijgedrag aan het einde van het wegvak, zoals
deze in de hier ingevoerde modellen gerealiseerd wordt,
is weergegeven in de flow-diagrammen van het programma
NETSIM. (Zie het programmadeel B1, pagina 32 t/m 39.)

Het vertrek uit de wegvakken wanneer de verkeers-
lichten groen zijn wordt in de programmadelen B2, B3 en B4
behandeld. De flow-diagrammen met het aanvullende
commentaar maken een uitvoerige beschrijving op deze
plaats overbodig.

-+-+-+-----

5. Flow-diagrammen.

Beschrijving van het programma NETSIM aan de hand van flow-diagrammen. (Vergelijk de listing in de bijlage.)



Alle variabelen en array's worden gedeclareerd en krijgen indien nodig een beginwaarde.

Zie data-input beschrijving.

Tevens worden de verkeerslichten gesynchroniseerd.

Iedere nieuwe volledige simulatie run begint bij RET met een leeg verkeersnetwerk.

Bij SIM begint ieder actietijdstip de volgende simulatie slag.

De klok welke het actietijdstip binnen de cyclus aangeeft wordt gelijkgezet.

Einde van een volledige simulatie run.

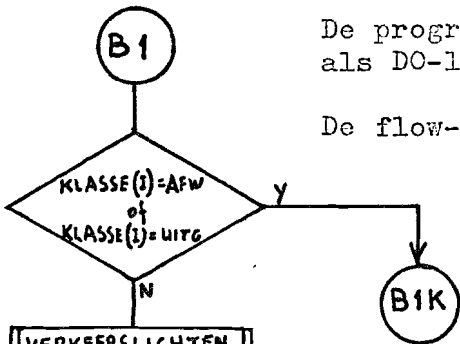
De resultaten-tellers worden op het tijdstip 1_1 op nul gezet.

Wanneer de simulatie al enkele cycli bezig is zal het netwerk niet leeg zijn zoals bij RET.

De tijdwijzers welke de actuele en eerstvolgende kolommen van de aankomstentabel aangeven worden gelijk gezet.

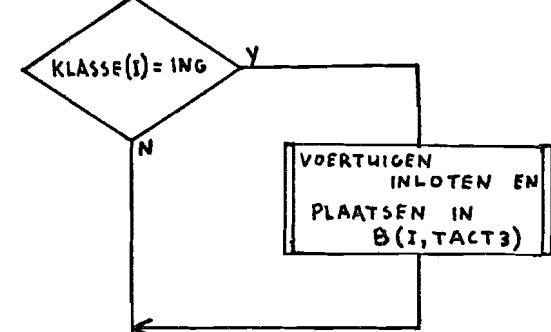
De programmadelen B1, B2, B3 en B4 worden elk als DO-loop uitgevoerd voor alle wegvakken.

De flow-diagrammen behandelen wegvak I.



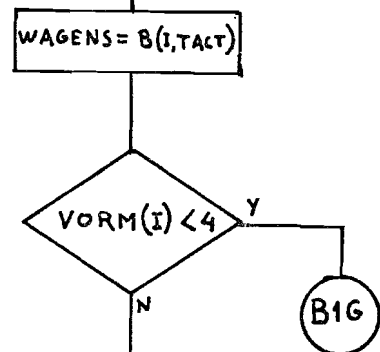
In B1 wordt de aankomst van voertuigen op de stopstreep en het plaats nemen in aankomstvakken, opstelvakken en files behandeld.

TFASE wordt vergeleken met GROENBEGIN(I,J) en ROODBEGIN(I,J) waarbij J de waarden 1, 2 en 3 doorloopt resp. geldend voor rechtsaf, rechtdoor en linksaf.

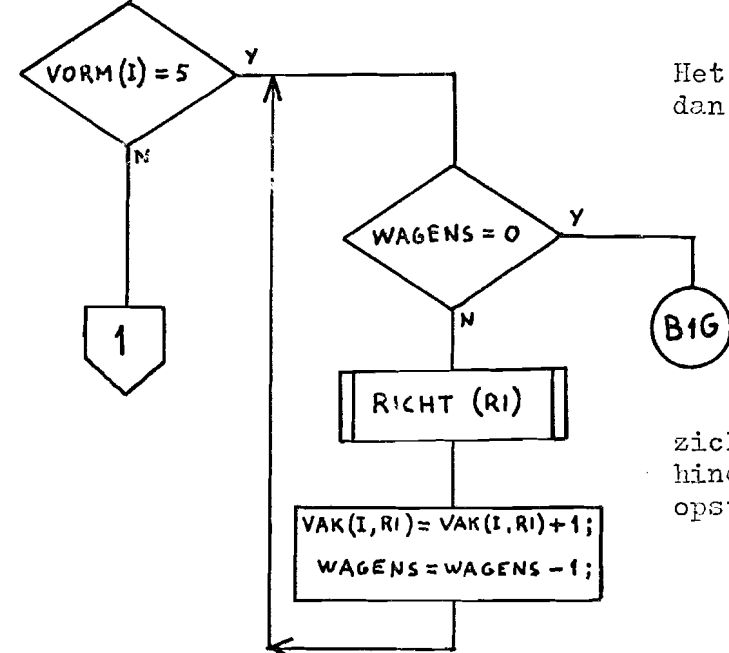


Aan de hand van de opgedrukte verkeersinput intensiteiten wordt voor alle rijstroken van het wegvak I geloot of er een auto binnen komt. Zo ja, dan wordt deze geplaatst in de aankomstentabel B(I,TACT3) zodat de auto over drie actietijdstippen de stopstreep bereikt.

Op het actuele actietijdstip bepaalt B(I,TACT) het aantal voertuigen dat aankomt en verwerkt moet worden.

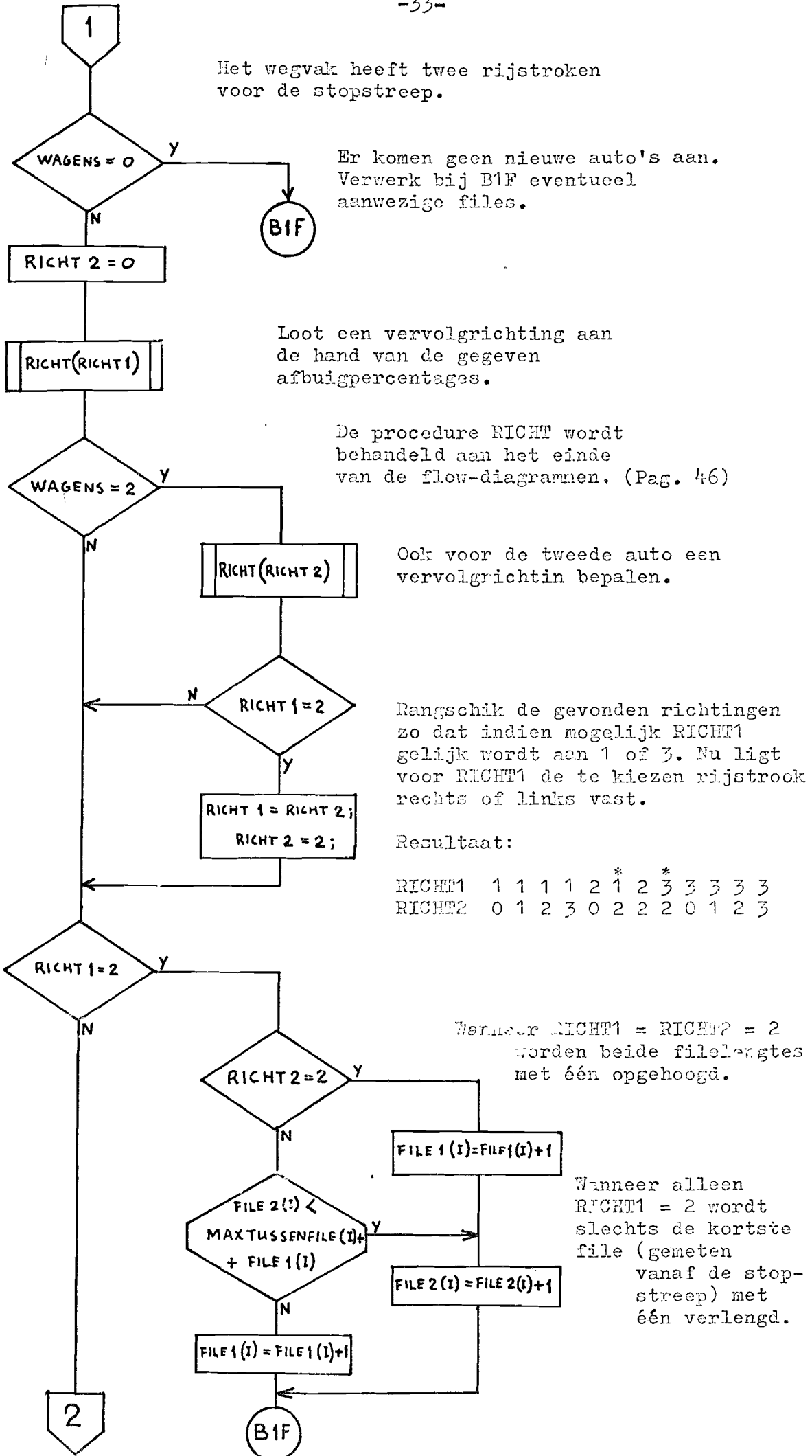


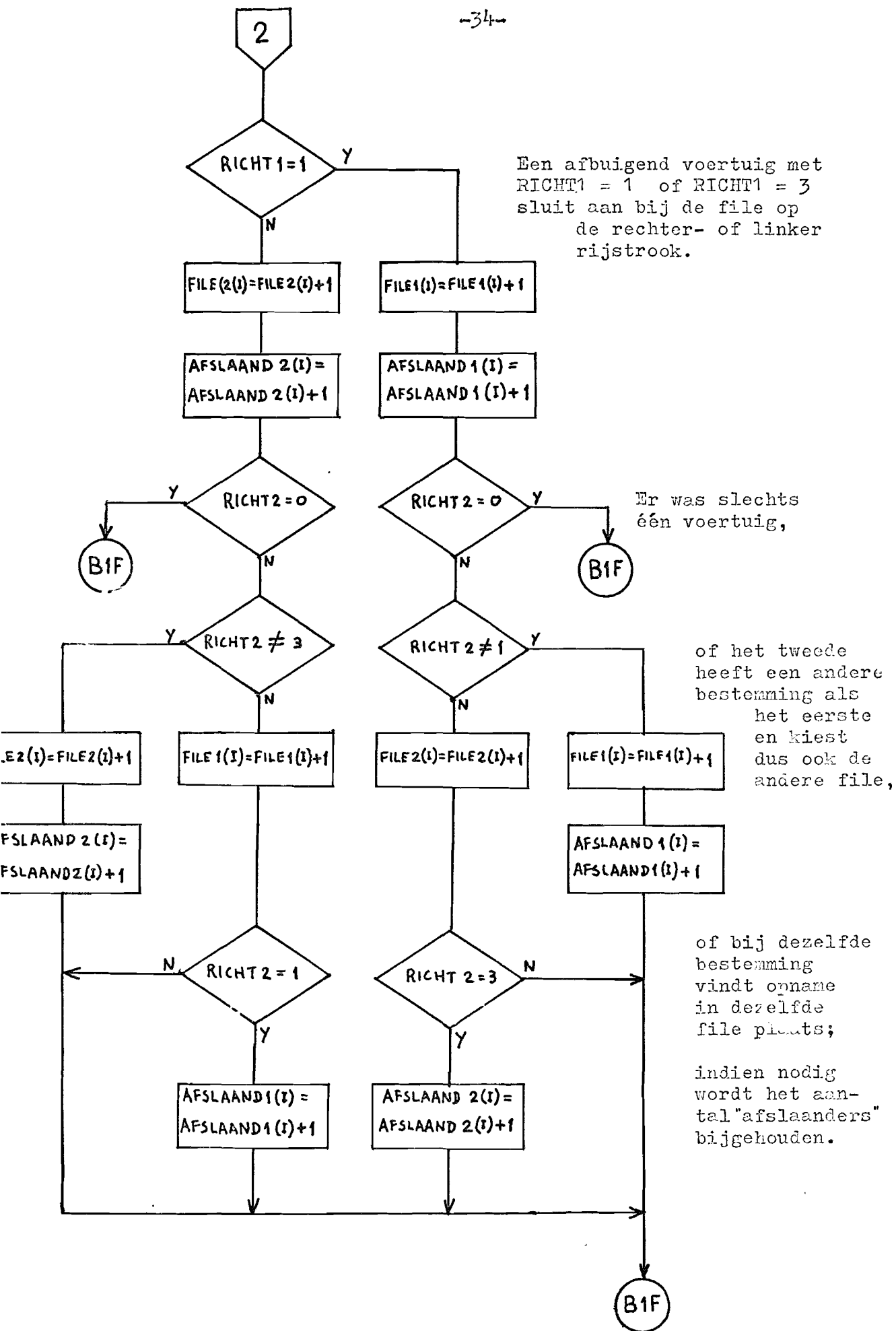
Het wegvak heeft één rijstrook voor de stopstreep.



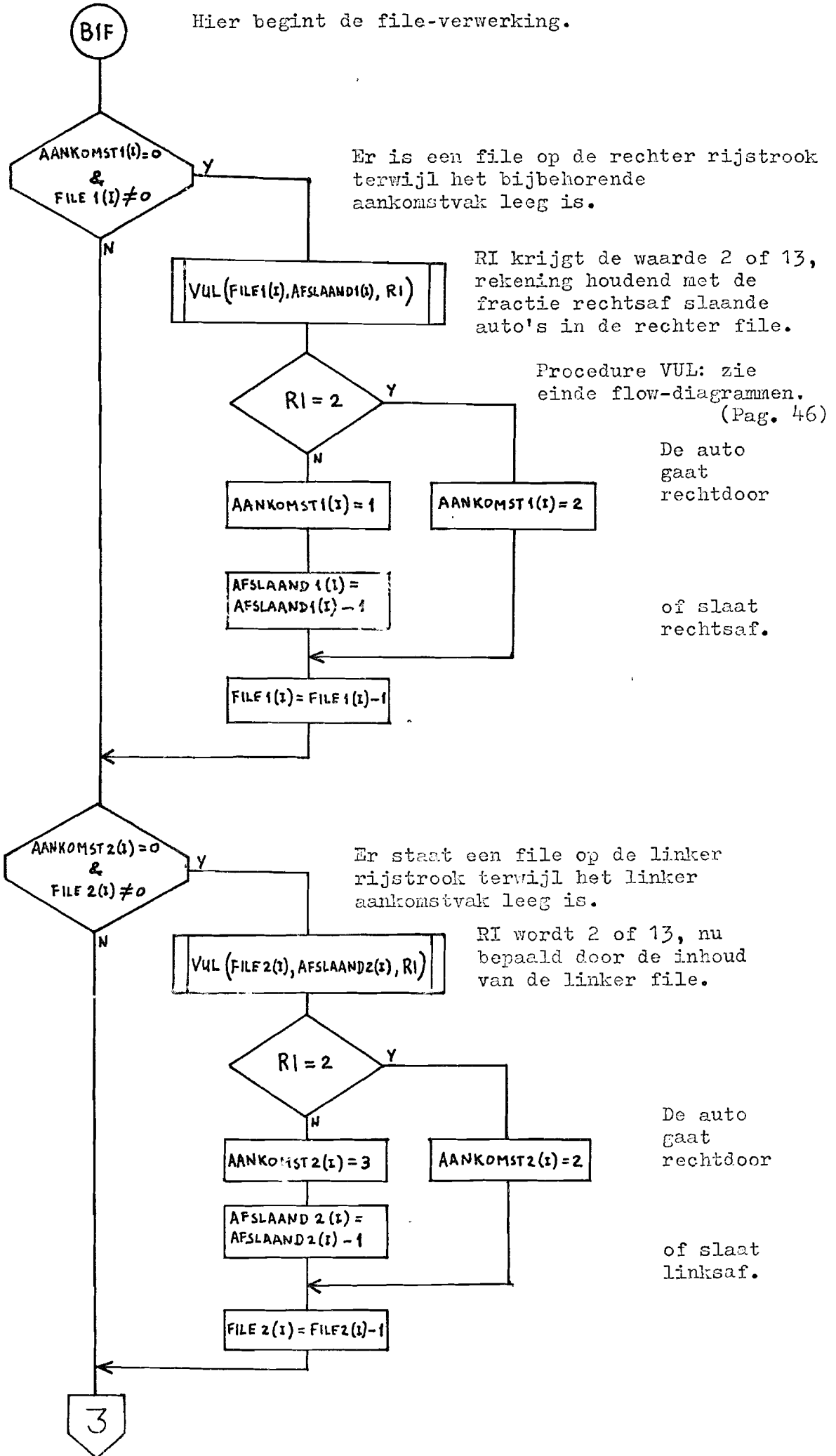
Het wegvak heeft meer dan twee rijstroken.

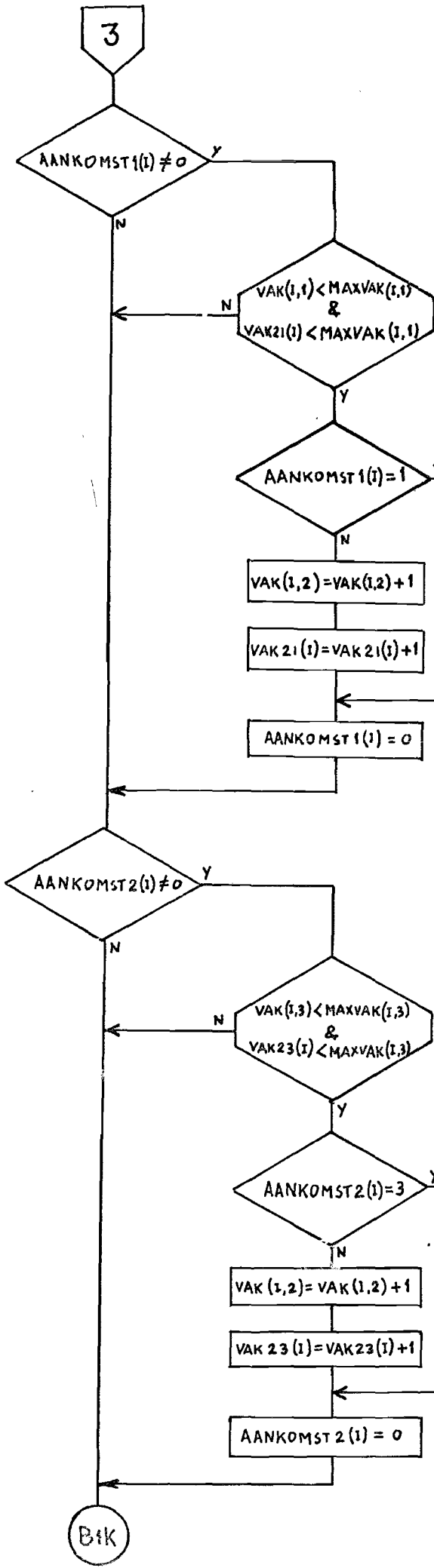
Voor alle aangekomen auto's wordt een vervolgrichting bepaald waarna ze zich zonder onderlinge hinder opstellen in de opstelvakken.





Hier begint de file-verwerking.





Een auto kan het vak AANKOMST1 slechts verlaten wanneer zowel het opstelvak voor rechtsaf als het opstelvak voor rechtdoor op de rechter rijstrook niet beide volledig gevuld zijn.

De auto gaat rechtsaf,

en wordt geplaatst in het rechter opstelvak,

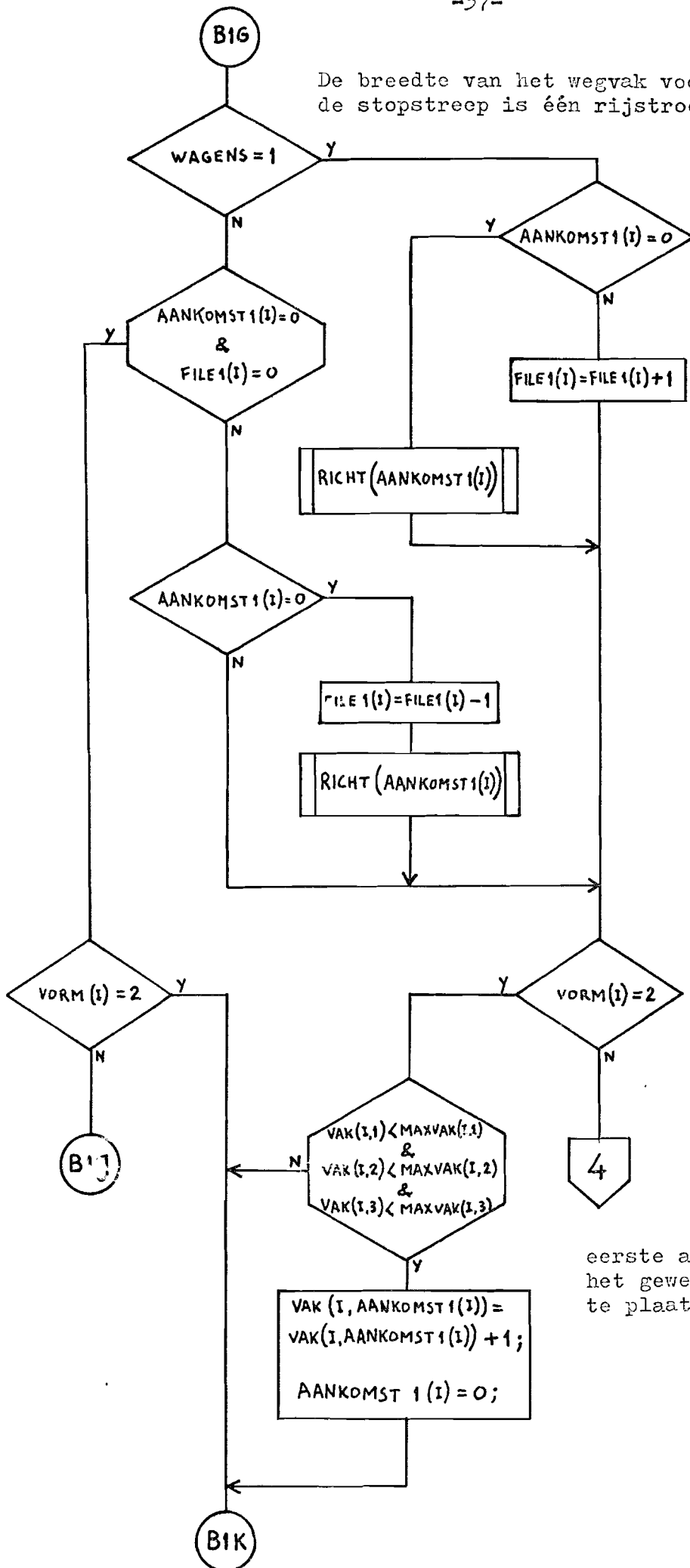
of in het opstelvak voor rechtdoor op de rechter rijstrook.

Om het vak AANKOMST2 te kunnen verlaten, mogen zowel het opstelvak voor linksaf als dat voor rechtdoor op de linker rijstrook niet beide volledig gevuld zijn.

De auto gaat linksaf,

en wordt in het linker opstelvak geplaatst,

of in het opstelvak voor rechtdoor op de linker rijstrook.



De breedte van het wegvak voor de stopstreep is één rijstrook.

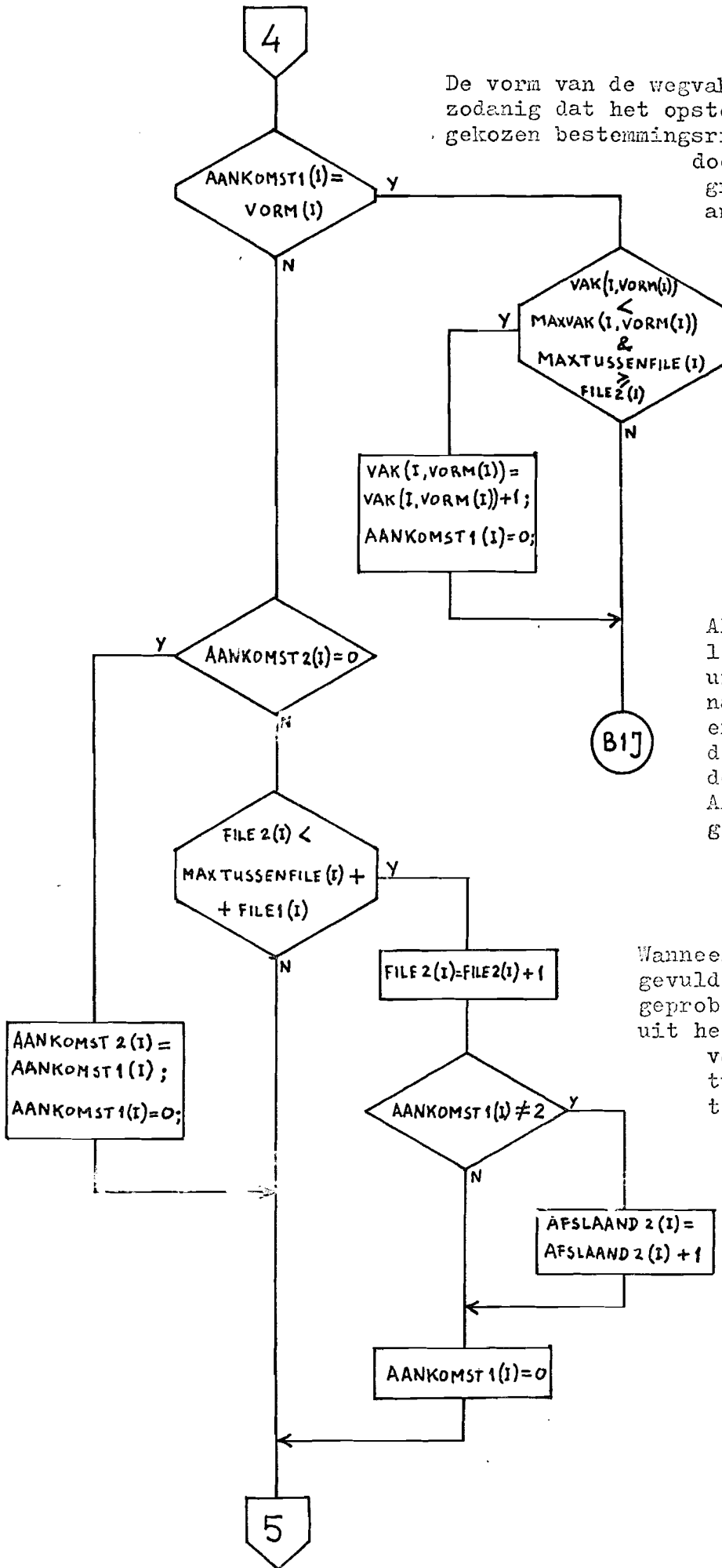
Er komt een auto aan,

welke geplaatst wordt in het eerste aankomstvak of de file ervoor.

Het eerste aankomstvak is leeg, en wordt gevuld uit de file die er voor staat.

Rechter en linker opstelvak zijn even groot,

en er wordt geprobeerd de auto uit het eerste aankomstvak in het gewenste opstelvak te plaatsen.

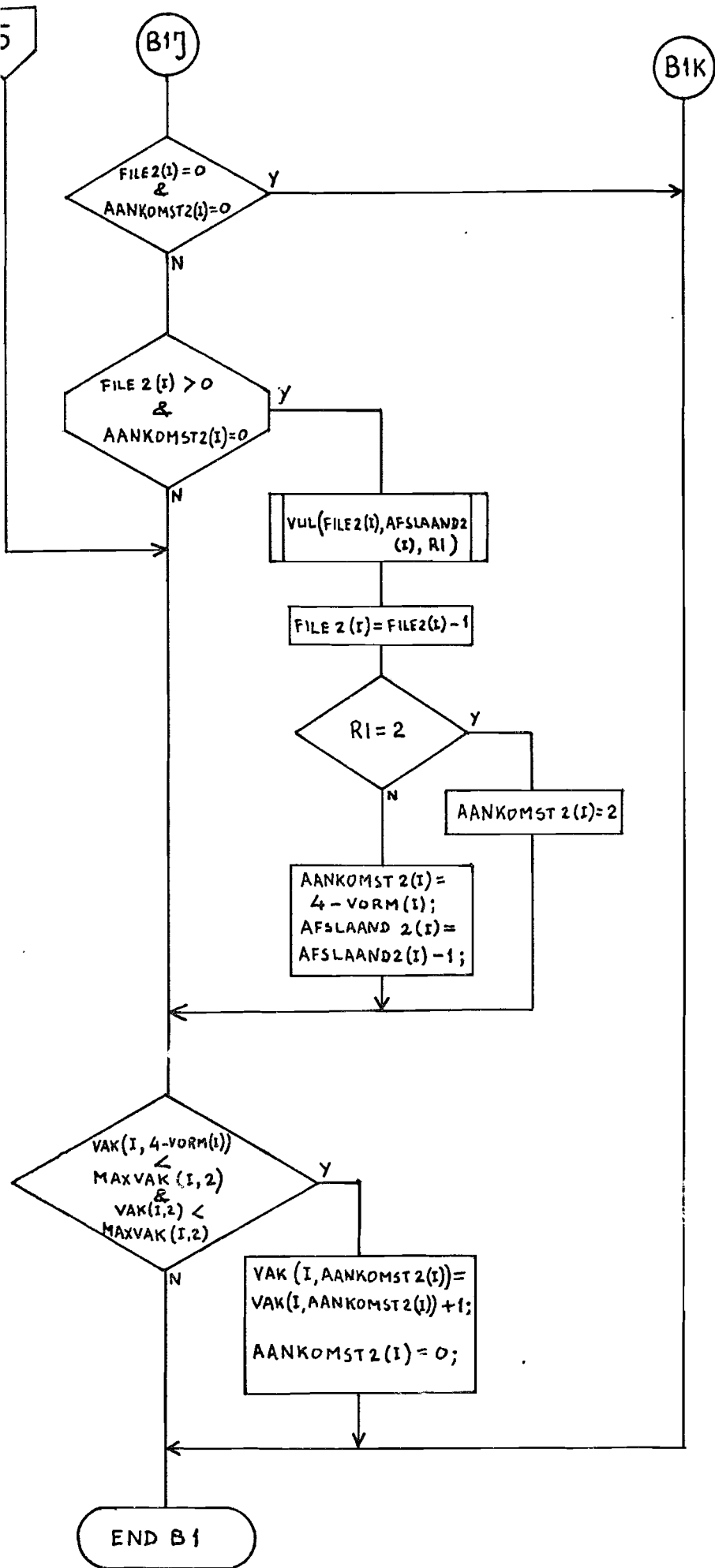


De vorm van de wegvak-uitmonding is zodanig dat het opstelvak voor de gekozen bestemmingsrichting (gegeven door AANKOMST1) groter is dan het andere opstelvak.

Indien mogelijk gaat de auto naar het gewenste opstelvak.

Als het vak AANKOMST2 leeg is gaat de auto uit het vak AANKOMST1 naar de tussenfile, en gelijktijdig vult de voorste auto uit de tussenfile het vak AANKOMST2 op met gelijke bestemming.

Wanneer AANKOMST2 al gevuld was wordt geprobeerd de auto uit het eerste aankomstvak in de tussenfile te plaatsen.



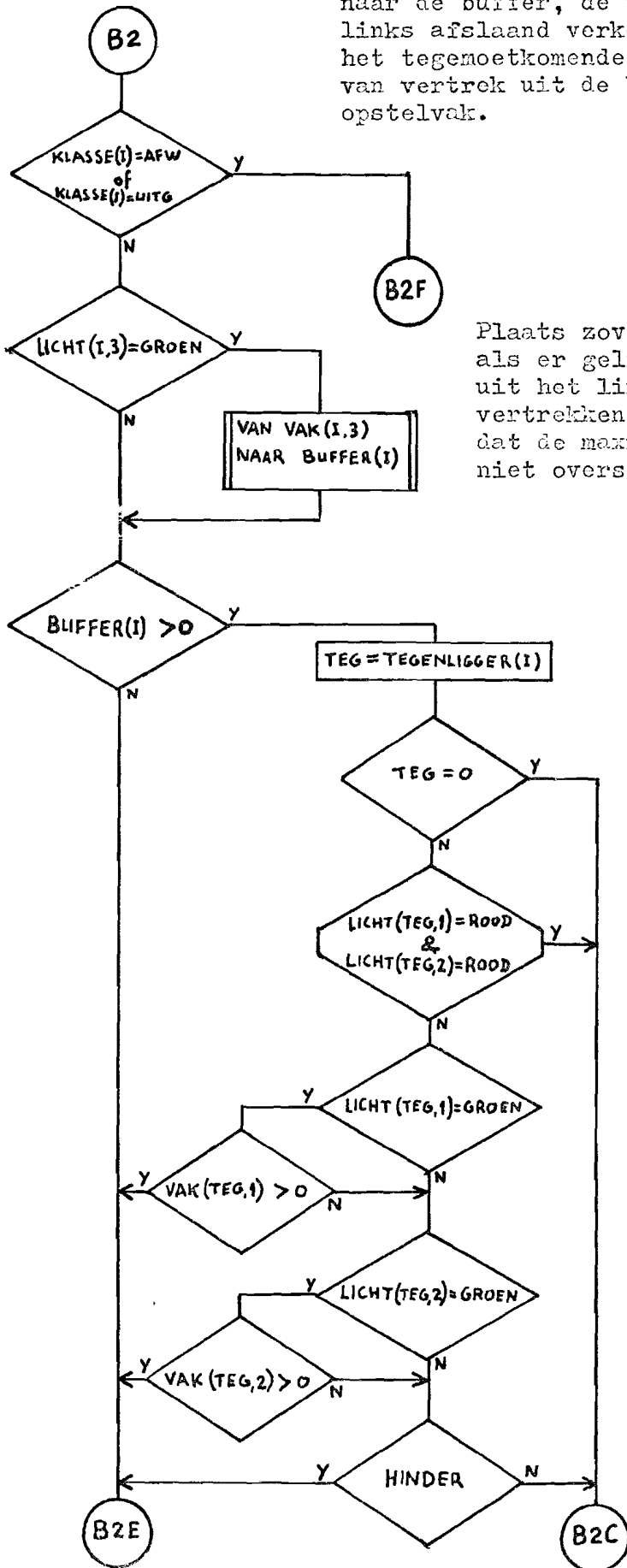
Er staat een tussenfile waaruit een voertuig het tweede aankomstvak kan opvullen.

De mogelijkheid is rechtdoor gaan,

of naar één zijde afslaan. (bepaald door VORM)

Indien mogelijk wordt de auto uit het tweede aankomstvak opgenomen door het gewenste opstelvak.

Het programmadeel B2 behandelt de overgang van voertuigen van het linker opstelvak naar de buffer, de test op hinder voor links afslaand verkeer veroorzaakt door het tegemoetkomende wegvak, en het toestaan van vertrek uit de buffer en het rechter opstelvak.



Plaats zoveel auto's in de buffer als er gelijktijdig naast elkaar uit het linker opstelvak kunnen vertrekken, onder de beperking dat de maximum inhoud van de buffer niet overschreden wordt.

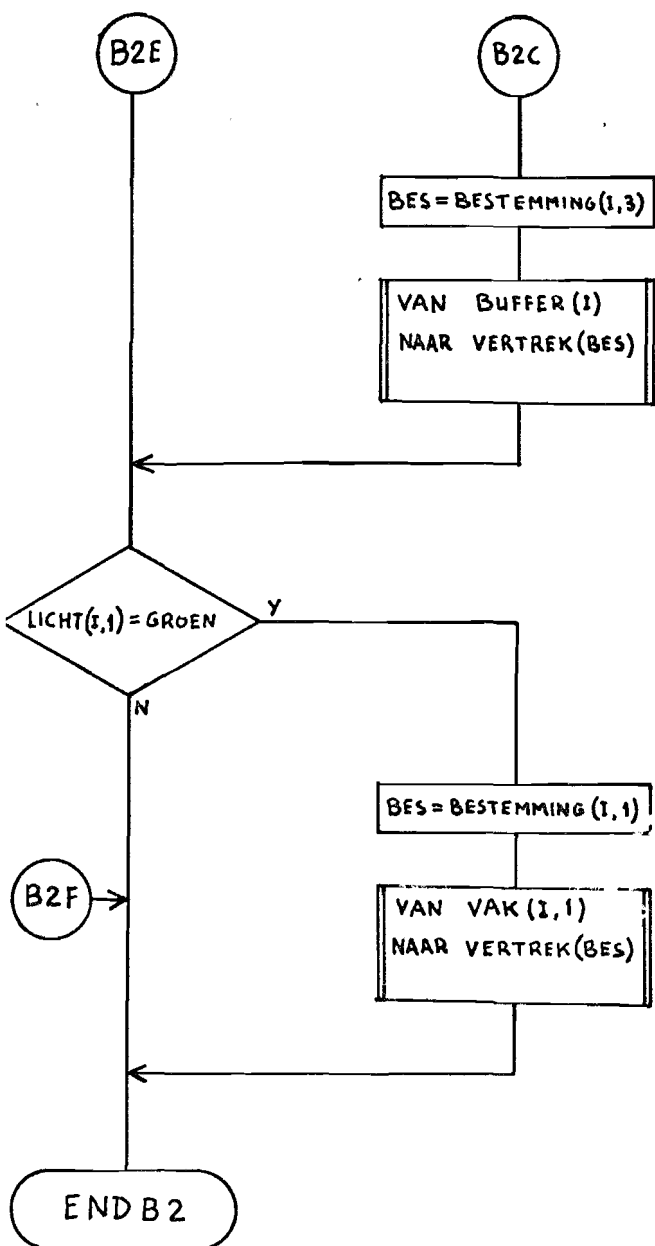
Geen tegemoetkomend wegvak, dus geen hinder.

Voor tegemoetkomend verkeer zijn de verkeerslichten rood, dus geen hinder.

Er is rechts-afslaand verkeer in het tegemoetkomende wegvak, dus hinder.

Er is rechtdoorgaand verkeer in het tegemoetkomende wegvak, dus hinder.

Onderzoek de aankomstvakken, files en de eerstvolgende drie posities van de aankomstentabel op hinder.



Er is geen hinder voor links afslaand verkeer. Geef zoveel auto's uit de buffer toestemming * tot vertrek als er naast elkaar het bestemming-wegvak kunnen inrijden.

Geef zoveel auto's uit het rechter opstelvak toestemming tot vertrek als er gelijktijdig naast elkaar het opstelvak kunnen verlaten.

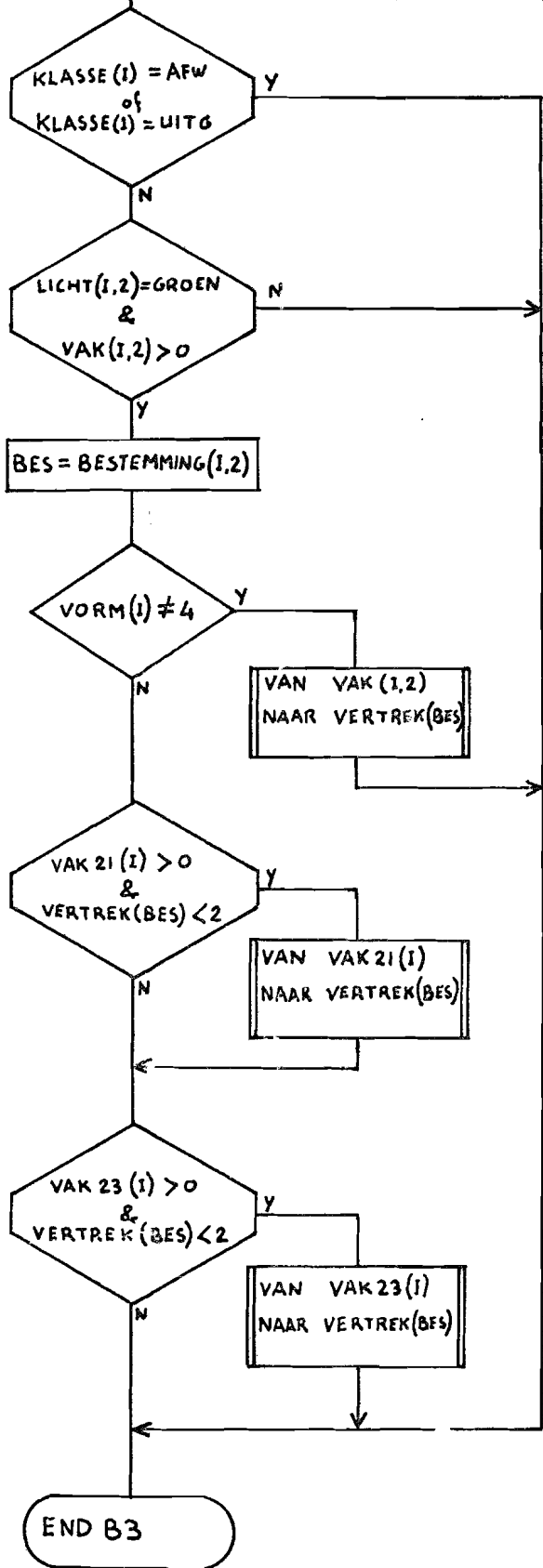
* Toestemming tot vertrek wil zeggen dat in array VERTREK op de plaatsen van de in te rijden wegvakken de vertrekkende auto's worden ingevuld.

Na afloop van de B2-loop is het verlenen van toestemming tot vertrek aan het afslaande verkeer uitgevoerd voor alle wegvakken van het netwerk.

Voor het rechtdoorgaande verkeer volgt nu B3. Doordat B2 klaar is wordt bereikt dat verkeer uit de buffer voorrang heeft op rechtdoorgaand verkeer met hetzelfde bestemmingswegvak.

B3

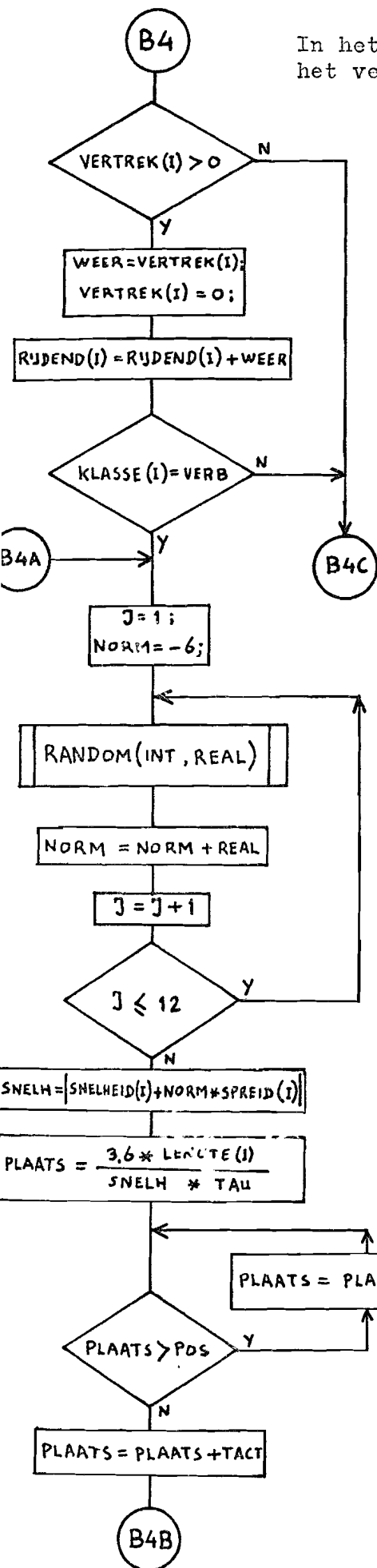
Het programmeeldeel B3 behandelt het rechtdoorgaande verkeer.



Wanneer het verkeerslicht groen is, en er ook auto's zijn die recht door gaan, wordt het bestemmingswegvak bepaald.

Toestemming tot vertrek verlenen onder de beperking dat het totale vertrek naar het bestemmingswegvak niet groter wordt dan de breedte van het opstelvak voor recht door waaruit vertrokken wordt.

In het programmadeel B4 wordt het vertrek definitief uitgevoerd.



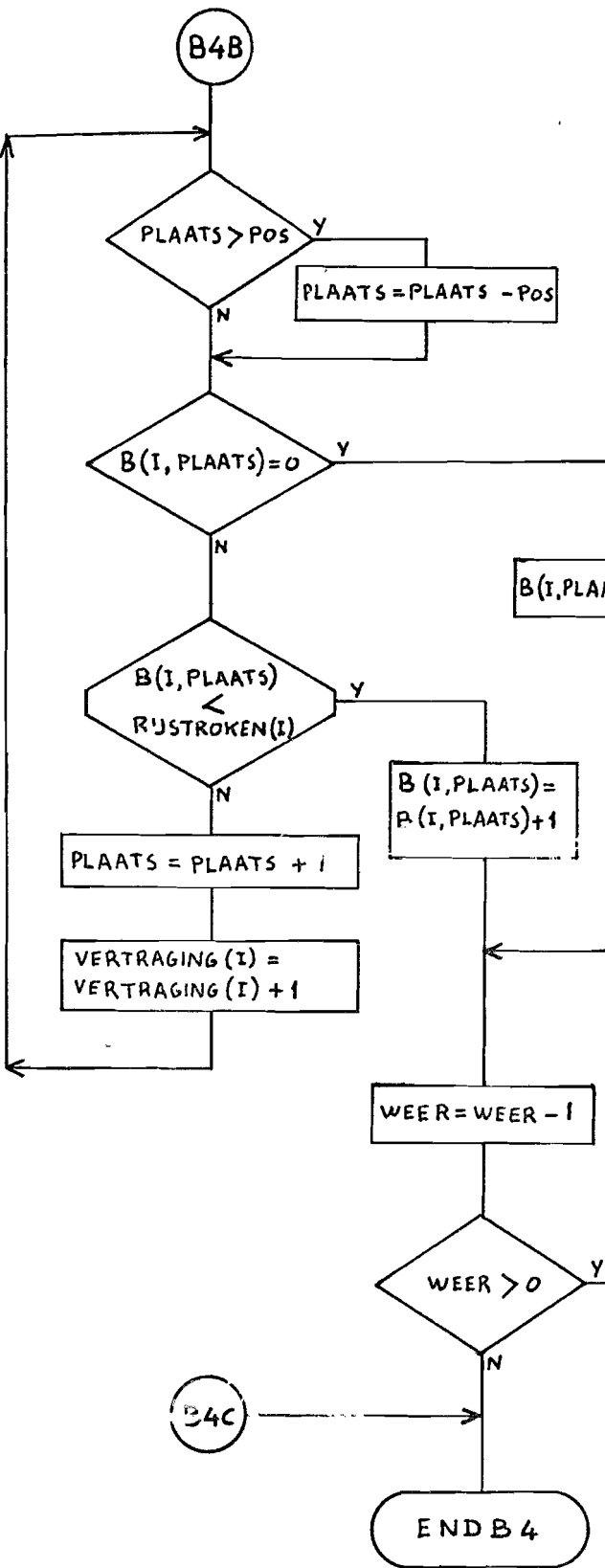
Alleen in uitgaande- en verbindingswegvakken is vertrek mogelijk. In het geval dat het bestemmingswegvak tot de klasse UITG behoort, wordt alleen het aantal rijdende voertuigen opgehoogd; bij verbindingswegvakken wordt voor elke vertrekkende auto een snelheid, en daarmee een rijtijd bepaald, en wordt de auto geplaatst in de aankomstentabel.

NORM is nu normaal verdeeld.
Gemiddelde waarde = 0
Spreading = 1

PLAATS geeft de rijtijd in actietijdslijpen.

PLAATS wordt gereduceerd wanneer deze groter is dan het aantal posities van de aankomstentabel, zodanig dat de fasetijd (binnen de verkeerslichten cyclus) van aankomst aan het einde van het wegvak gehandhaafd blijft.

PLAATS wordt aangepast aan de actuele tijd TACT.



Als PLAATS het aantal posities van de aankomstentabel overschrijdt moet er eerst POS afgetrokken worden.

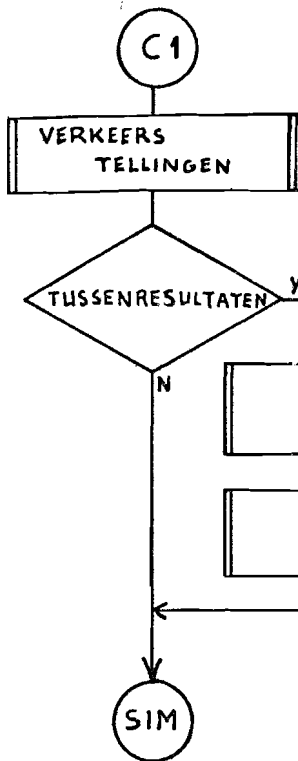
De gewenste positie is leeg waardoor plaatsing zonder meer mogelijk is.

De bezetting van de gewenste positie is kleiner dan het aantal rijstroken, dus plaatsing is mogelijk;

in het andere geval wordt de tijd van aankomst één actietijdstip vertraagd, waarna opnieuw geprobeerd wordt de auto te plaatsen.

De gehele actie moet voor ieder vertrekkend voertuig uitgevoerd worden.

Wanneer de B4-loop voor alle wegvakken is uitgevoerd wordt het programma voortgezet bij C1.



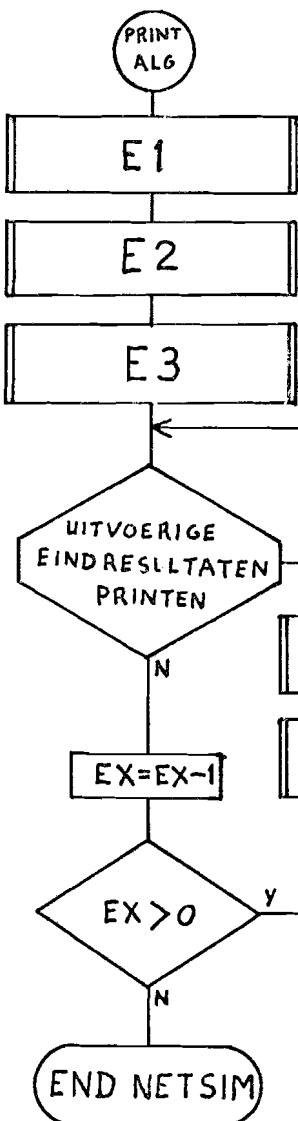
Aan het einde van ieder actietijdstip wordt het aantal wachtende voertuigen per wegvak geteld, en wordt de totale wachttijd berekend.

In het geval dat tussenresultaten gevraagd worden geven

D1: alle niet lege aankomstentabellen van de verbindingswegvakken.

D2: de overige verkeersgegevens van alle ingaande- en verbindingswegvakken.

Terug naar SIM voor het volgende actietijdstip.



Wanneer de simulatie beëindigd is worden de resultaten afgedrukt voor:

E1: de verbindingswegvakken

E2: de ingaande wegvakken

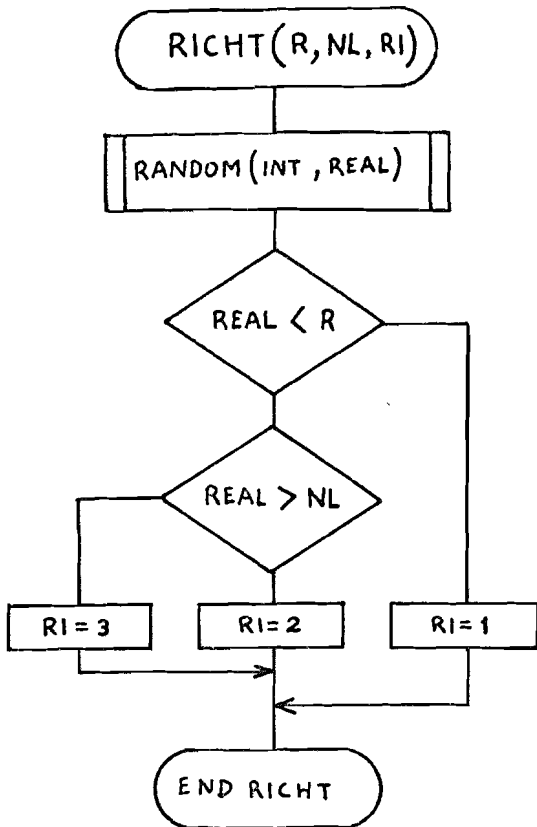
E3: de uitgaande wegvakken.

Worden uitvoerige simulatie resultaten gevraagd dan leveren

F1: het verloop van de filelengte tijdens de gehele simulatieperiode.

F2: de aankomstverdeling binnen de cyclustijd van voertuigen op de stopstreep.

Indien nodig wordt een nieuwe simulatie run uitgevoerd.



De procedure RICHT levert een bestemmingsrichting RI uitgaande van de kans op rechts R en de kans op niet-links NL

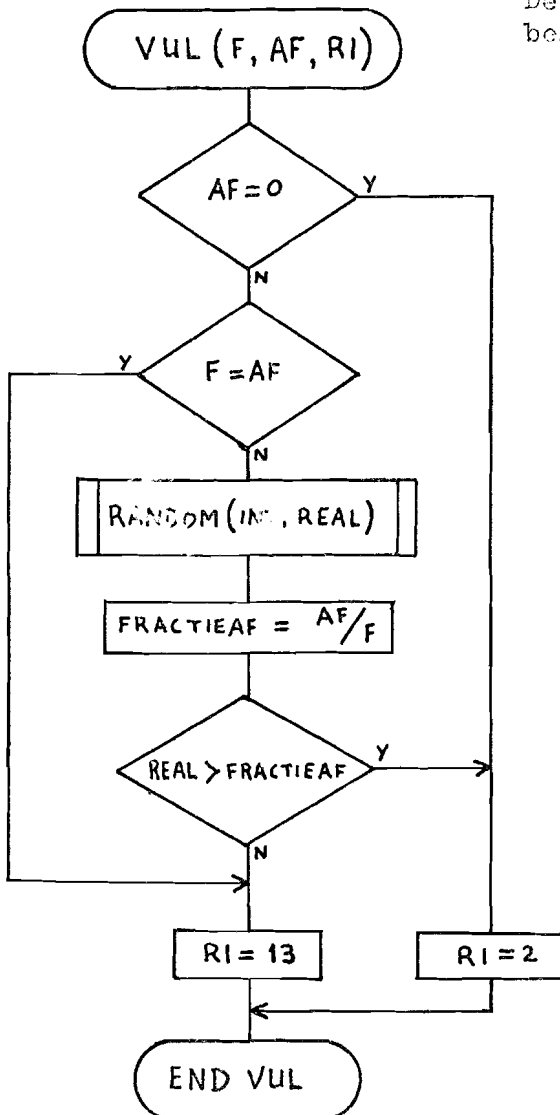
Uit het interval (0,1) wordt random een getal REAL getrokken.

Afhankelijk van REAL wordt:

RI = 1 (rechtsaf)

RI = 2 (rechtdoor)

RI = 3 (linksaf)



De procedure VUL levert een bestemmingsrichting RI voor een voertuig uit een file met twee mogelijke vervolgrichtingen. Parameters zijn de filelengte F, en het aantal afslaande voertuigen AF.

Uit het interval (0,1) wordt random een getal REAL getrokken.

Afhankelijk van REAL wordt:

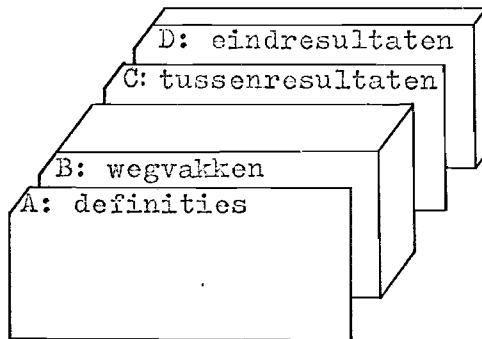
RI = 2 (rechtdoor)

RI = 13 (afslaand)

De betekenis van RI = 13 (rechtsaf of linksaf) is bekend op de plaats van aanroep.

6. Data-input beschrijving

Plaats alle getallen zo, dat het laatste cijfer in de hoogst genummerde toegewezen kolom staat. Niet-significante nullen behoeven niet geponst te worden.



De in te voeren gegevens worden op de volgende wijze gegroepeerd en geponst in kaarten als data toegevoegd aan het programma.

- A: definities
- B: wegvakken
- C: tussenresultaten
- D: eindresultaten

A. De definitie-gegevens zijn nodig om de grootte van het netwerk en een aantal belangrijke tijdgrootheden aan te geven. Al deze gegevens worden als volgt geponst in één ponskaart.

Kolon	Omschrijving
49 - 52	PRE is het aantal inleidende simulatie-cycli, voordat de eigenlijke simulatie begint. Doel is het netwerk te vullen met voertuigen. $PRE \in \{0, 1, 2, \dots\}$
53 - 56	EM is het aantal malen dat het totale programma moet worden uitgevoerd. Hierbij start de random-generator iedere keer bij een ander getal.
57 - 60	KR is het aantal kruispunten waaruit het netwerk bestaat.

Kolom	Omschrijving
61 - 64	AANTAL is het hoogst toegewezen volgnummer aan de wegvakken.
65 - 68	CSEC is de tijdsduur in seconden van één verkeerslichten-cyclus.
69 - 72	CTAU is het aantal actietijdstippen waarin één verkeerslichten-cyclus verdeeld wordt. Kies CTAU zo dat het quotiënt van CSEC en CTAU precies of bij benadering gelijk is aan 2.
73 - 76	CSIM is het aantal te simuleren cycli, waaraan er dus PRE vooraf gaan.
77 - 80	POS is het aantal posities van de aankomstentabel. Minimale waarde: CTAU. Kies b.v. POS = 100

- B. Voor alle wegvakken welke in het netwerk voorkomen moet een aparte ponskaart gebruikt worden. Om het einde van de wegvak-kaarten aan te geven moet de laatste wegvak-kaart gevolgd worden door een afsluitkaart welke een 0 bevat in kolom 3. De wegvak-kaarten zijn als volgt samengesteld:

Kolom	Omschrijving
1 - 3	Het volgnummer i van het wegvak. Alle n wegvakken welke bij het te simuleren netwerk betrokken zijn worden oplopend genummerd van 1 tot AANTAL waarbij geldt: $n \leq \text{AANTAL}$.
5	KLASSE van het wegvak. Wegvakken die niet in het netwerk voorkomen hebben KLASSE = 0. Uitgaande wegvakken behoren tot KLASSE = 1; ingaande wegvakken tot KLASSE = 2 en verbindingswegvakken tot KLASSE = 3.
7	Het aantal RIJSTROKEN van het wegvak. Dit aantal komt overeen met het aantal voertuigen dat per actietijdstip een bepaald punt kan passeren. Voor veel wegvakken waarin het verkeer gehinderd wordt door tegenliggers geldt als breedte 1 rijstrook hoewel de mogelijkheid van inhaal wel bestaat.
74 - 76	Het vertrek-kruispunt VAN. Voor wegvakken uit KLASSE = 2 welke het vertrek-kruispunt buiten het netwerk hebben geldt VAN = 0.
78 - 80	Het bestemmings-kruispunt NAAR. Voor wegvakken uit KLASSE = 1 welke het bestemmings-kruispunt buiten het netwerk hebben geldt NAAR = 0.

De kaarten van de ingaande- en verbindingswegvakken (resp. KLASSE = 2 en KLASSE = 3) bevatten verder nog de volgende gegevens:

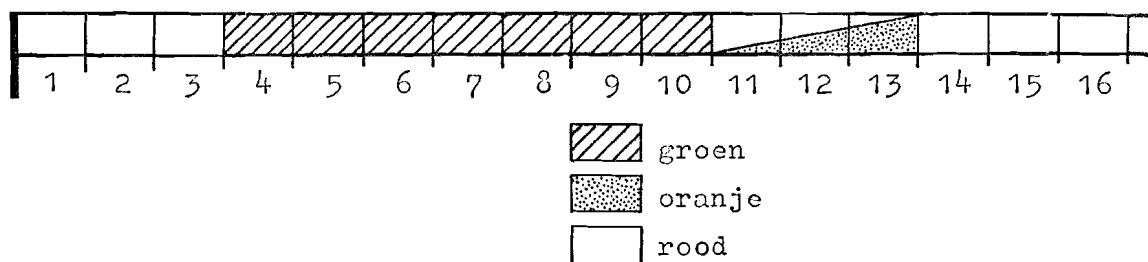
KOLOM	Omschrijving						
9	De VORM van de wegvak-uitmonding. Wegvakken met meer dan 2 rijstroken hebben VORM = 5. Wegvakken met 1 of 2 rijstroken, welke twee opstelvakken hebben voor rechtdoorgaand verkeer (al dan niet tevens in gebruik door afslaand verkeer) hebben VORM = 4. Wegvakken met 1 rijstrook en één opstelvak voor rechtdoorgaand verkeer hebben: VORM = 2 als de opstelvakken voor rechts- en linksaf even groot zijn. VORM = 1 als het opstelvak voor rechtsaf groter is dan dat voor linksaf. VORM = 3 als het opstelvak voor rechtsaf kleiner is dan dat voor linksaf.						
11 - 13	Het volgnummer van het bestemmingswegvak voor links afslaand verkeer.						
15 - 17	Het volgnummer van het bestemmingswegvak voor rechtdoor gaand verkeer.						
19 - 21	Het volgnummer van het bestemmingswegvak voor rechts afslaand verkeer.						
23 - 25	Het volgnummer van het op het bestemmingskruispunt tegemoetkomend wegvak. Het verkeer hieruit kan hinder opleveren voor links afslaand verkeer.						
26 - 28	Het percentage RECHTSAFslaand verkeer aan het einde van het wegvak.						
29 - 31	Het percentage NIETLINKSAFslaand verkeer aan het einde van het wegvak. Altijd geldt: NIETLINKSAF \geq RECHTSAF.						
32 - 34 35 - 37 38 - 40	<table> <tr> <td rowspan="3">}</td> <td rowspan="3">Het GROENBEGINTIJDSTIP* in seconden van het verkeerslicht voor:</td> <td rowspan="3">}</td> <td>linksaf</td> </tr> <tr> <td>rechtdoor</td> </tr> <tr> <td>rechtsaf</td> </tr> </table>	}	Het GROENBEGINTIJDSTIP* in seconden van het verkeerslicht voor:	}	linksaf	rechtdoor	rechtsaf
}	Het GROENBEGINTIJDSTIP* in seconden van het verkeerslicht voor:				}	linksaf	
						rechtdoor	
		rechtsaf					
41 - 43 44 - 46 47 - 49	<table> <tr> <td rowspan="3">}</td> <td rowspan="3">Het ROODBEGINTIJDSTIP* in seconden van het verkeerslicht voor:</td> <td rowspan="3">}</td> <td>linksaf</td> </tr> <tr> <td>rechtdoor</td> </tr> <tr> <td>rechtsaf</td> </tr> </table>	}	Het ROODBEGINTIJDSTIP* in seconden van het verkeerslicht voor:	}	linksaf	rechtdoor	rechtsaf
}	Het ROODBEGINTIJDSTIP* in seconden van het verkeerslicht voor:				}	linksaf	
						rechtdoor	
		rechtsaf					
51 - 52	Het aantal voertuigen MAMBUF dat zich maximaal in de buffer op het kruispunt kan opstellen.						
60	BREEDTE(BUF) is het aantal voertuigen dat per actietijdstip de buffer kan verlaten.						

* zie de volgende pagina.

Het GROENBEGINTIJDSTIP en het ROODBEGINTIJDSTIP kunnen afgelezen worden uit het fase-diagram van het betreffende wegvak. Deze moet gegeven zijn op een tijdsbasis in seconden.

De oranje-fase wordt bij groen gerekend.

Voorbeeld.



GROENBEGINTIJDSTIP = 4

ROODBEGINTIJDSTIP = 14

Wegvakken met VORM = 5 hebben in de kolommen 54, 56 en 58 het aantal voertuigen dat bij groen verkeerslicht per actietijdstip de opstelvakken kan verlaten.

Kolom	Omschrijving
54	BREEDTE(3) van het opstelvak voor linksaf.
56	BREEDTE(2) van het opstelvak voor rechtdoor.
58	BREEDTE(1) van het opstelvak voor rechtsaf.

Wegvakken met VORM < 5 krijgen in plaats van de hierboven vermelde breedtes de inhouden van de opstelvakken gemeten in voertuigen. De indeling van de kaart wordt dan:

Kolom	Omschrijving
54 - 55	MAXL is de maximale inhoud van het opstelvak voor links afslaand verkeer.
57 - 58	MAXR is de maximale inhoud van het opstelvak voor rechts afslaand verkeer.

Wegvakken van KLASSE = 2 krijgen het inkomende verkeer aangeboden zodat op de kaart toegevoegd moet worden:

Kolom	Omschrijving
61 - 65	De intensiteit van inkomend verkeer in voertuigen per uur. Als maximum geldt één voertuig per actietijdstip per rijstrook.

Wegvakken uit KLASSE = 3 liggen geheel binnen het netwerk, vandaar dat de volgende gegevens toegevoegd moeten worden.

Kolom	Omschrijving
62 - 65	De lengte van het wegvak in meters (gemeten van het middelpunt van het vertrek-kruispunt tot het middelpunt van het bestemmingskruispunt.
66 - 68	De gemiddelde waarde van de gemiddelde snelheden waarmee de voertuigen de totale lengte van het wegvak afleggen in het geval de doorgang op het bestemmingskruispunt niet belemmerd wordt. (In km/uur)
69 - 71	De spreiding in de gemiddelde snelheden in km/uur.

- C. Eén kaart is voldoende om de gegevens betreffende de uit te printen tussenresultaten op te geven. De indeling is als volgt:

Kolom	Omschrijving
1 - 4	NG is het aantal malen per simulatie-run dat de gehele netwerkstatus uitgeprint moet worden. Wanneer NG = 0 blijft de rest van de kaart leeg.
5 - 8	PRC geeft de cyclus aan dat voor het eerst tussenresultaten gewenst zijn.
9 - 12	PRT geeft de fase-tijd in de hiervoor bepaalde cyclus PRC, dat voor het eerst tussenresultaten gewenst zijn.
13 - 16	PRG is het aantal opvolgende actietijdstippen dat tussenresultaten geprint worden.
17 - 20	PRS is het aantal daarop volgende actietijdstippen dat <u>geen</u> tussenresultaten geprint worden.

Opn. Tijdens executie van het programma worden PRG en PRS herhaaldelijk doorlopen totdat NG maal tussenresultaten afgedrukt zijn.

- D. Eindresultaten in uitvoerige, gedeeltelijk grafische vorm, kunnen opgevraagd worden per tweetal wegvakken. De volgnummers van alle gewenste paren wegvakken worden, elk met tussenruimte van twee lege kolommen achter elkaar gepost in de benodigde ponskaarten. Na het laatste wegvak nummer volgt een 0 om het einde aan te geven.

7. Output beschrijving van de simulatie resultaten.

Tijdens en na afloop van de simulatie kunnen diverse resultaten uitgeprint worden. Deze data-uitvoer valt uit-
één in drie delen.

- a) Tussenresultaten in de vorm van de netwerkstatus worden indien gewenst afgedrukt voor een aantal op te geven actietijdstippen.
- b) Het eindresultaat van de simulatie wordt in compacte vorm afgedrukt voor alle wegvakken welke deel uitmaken van het netwerk.
- c) Een uitgebreid overzicht van het verloop van enkele karakteristieke verkeers grootheden tijdens de hele simulatie-periode wordt afgedrukt van elk gewenst tweetal wegvakken.

sub a. Wanneer tussenresultaten gewenst zijn moet een data-kaart met zes getallen ingelezen worden, waarmee wordt opgegeven welke actietijdstippen de netwerkstatus afgedrukt moet worden. Deze actietijdstippen zijn periodiek verdeeld over de simulatie run, zoals blijkt uit onderstaande toelichting op de inhoud van de data-kaart. Deze bevat achtereenvolgens:

- NG het aantal malen dat tussenresultaten uitgeprint moeten worden.
- PRC en PRT de cyclus en de fasetijd binnen die cyclus dat voor het eerst uitvoer gewenst is.
- PRG het aantal opvolgende actietijdstippen waarop tussenresultaten afgedrukt worden.
- PRS het aantal daaropvolgende actietijdstippen dat geen tussenresultaten afgedrukt worden.

Totdat NG bereikt is worden PRG en PRS afwisselend doorlopen

Voorbeeld. Een data-kaart met als inhoud: 6 3 1 2 5
levert de volgende actietijdstippen:

3_1 3_2 3_8 3_9 3_15 3_16

(De notatie i_j betekent: cyclus i ,
actietijdstip j .)

De gehele netwerkstatus wordt gegeven, zowel wat betreft het stilstaande verkeer in de opstelvakken, buffers, files, enz. als het rijdende verkeer in de verbindings wegvakken. Van deze laatste wordt eerst de aankomstentabel afgedrukt, waarbij de rijen van de wegvakken waarin geen rijdend verkeer aanwezig is, overgeslagen worden. De eerste kolom van de aankomstentabel bevat het aantal auto's dat het volgende actietijdstip de stopstreep zal bereiken; elke volgende kolom de auto's welke één actietijdstip later aankomen.

Na de aankomstentabel volgt voor alle verbindings- en ingaande wegvakken in tabelvorm een uitgebreide opgave van hun status. De er boven afgedrukte kop bestaat uit de hieronder verklaarde benamingen.

NR	Volgnummer van het wegvak.
LICHT 321	Kleur van het verkeerslicht (R of G) resp. voor linksaf, rechtdoor en rechtsaf.
TABEL 123	Onder de cijfers 123 staan de drie eerstvolgende posities van de aankomstentabel.
STILSTAAND	Aantal wachtende voertuigen voor de stopstreep.
BUF	Inhoud van de buffer van het betreffende wegvak op het kruispunt.
VAK 3,2,1	Inhoud van de opstelvakken voor de stopstreep, resp. voor linksaf, rechtdoor en rechtsaf.
VAK23,21	De verdeling van de inhoud van VAK 2 over de linker- en rechter rijstrook.
AK 2,1	Bestemming in de vakken AANKOMST2 en AANKOMST1; afgedrukt worden < linksaf, rechtdoor en > rechtsaf.
FILE2,1	Inhoud van file2 en file1
AFSL2,1	Aantal afslaannde auto's in file2 resp. file1.
WACHTTIJD	Gesommeerde wachttijd vanaf cyclus: 1, actietijdstip: 1. (in voertuig-actietijdstippen)
VERTRAGING	Opgelopen vertraging in het wegvak ten gevolge van de plaatsings procedure in de aankomstentabel. (in voertuig-actietijdstippen)

VERTROKKEN	De uit het wegvak, inclusief de buffer vertrokken voertuigen vanaf cyclus: 1, actietijdstip: 1.
STOPS	Het aantal wachtende voertuigen, gesommeerd op de actietijdstippen dat het rechtdoorlicht groen wordt.
RIJDEND	Het aantal rijdende voertuigen in de verbindings wegvakken.

sub b.

Het overzicht van de simulatie resultaten valt uiteen in een algemeen gedeelte waarin stops en wachttijden voor het gehele netwerk gegeven worden, en een opsomming van de simulatie resultaten van alle wegvakken gegroepeerd naar de klassen: verbindend, ingaand en uitgaand.

Door het discrete karakter van de simulatie en de daarmee samenhangende stapsgewijze verplaatsing van de voertuigen kan het stoppen voor de stopstreep niet direct waargenomen worden. Om deze reden wordt hier een 'stop' gedefinieerd als de aanwezigheid van een stilstrand voertuig voor de stopstreep op het moment dat het rechtdoorverkeerslicht van het betreffende wegvak groen wordt. Door deze definitie dragen voertuigen welke tijdens de groenfase aangekomen, gestopt en daarna weer vertrokken zijn, niet bij tot het aantal 'stops'. Wel dragen bij de voertuigen die tijdens de roodfase aankomen, en er geldt dat voertuigen welke meerdere cycli moeten wachten voordat vertrek mogelijk is evenzovaak meetellen in het aantal 'stops'.

Verklaring van de boven de tabellen afgedrukte kop.

INGEKOMEN	Het aantal voertuigen dat een bepaald wegvak ingekomen is. Dit komt overeen met de som van het aantal voertuigen dat aan het einde van de simulatie nog rijdt, nog staat te wachten voor de stopstreep en in de buffer, en al vertrokken is naar een ander wegvak.
UITGEGAAN	Het aantal voertuigen dat een wegvak (inclusief de bijbehorende buffer voor linksaf) heeft verlaten.
MAXIMUM FILE	Het grootste aantal voertuigen dat (wachtend) voor de stopstreep opgetreden is, geteld op het moment dat het verkeerslicht voor recht door groen werd.

MINIMAAL BENODIGDE CYCLI	Quotiënt van MAXIMUM FILE en het grootste aantal voertuigen dat in één cyclus het wegvak verlaten heeft.
FILE NA AFLOOP SIMULATIE	Momentopname; het aantal wachtenden op het einde van de simulatierun.
'STOPS'	Het aantal stops zoals hiervoor gedefinieerd, gesommeerd over de hele simulatierun.
WACHTTIJD STILSTAAND	Het aantal stilstaande voertuigen per wegvak gesommeerd over alle actietijdstippen.
VERTRAGING RIJDEND	Het aantal actietijdstippen vertraging dat het verkeer oploopt ten gevolge van de plaatsings procedure in de aankomstentabel. Deze vertraging veroorzaakt een afwijking op de toegepaste snelheids- cq. rijtijdenverdeling.
WACHTTIJD TOTAAL	Som van WACHTTIJD STILSTAAND en VERTRAGING RIJDEND.

sub c.

Uitgebreide simulatieresultaten worden afgedrukt per tweetal wegvakken. Sinnvolle combinaties kunnen zijn: twee wegvakken welke samen komen op een kruispunt en dezelfde groen-rood verdeling hebben, (tegenoet komend verkeer) of twee wegvakken welke in elkaars verlengde liggen en een doorgaande verkeersstroom verwerken.

Allereerst wordt gegeven: het verloop van het aantal wachtenden in de tijd (ook hier altijd gemeten op het groenbegintijdstip van het verkeerslicht voor recht door) en het aantal vertrokken voertuigen per groen-rood periode. Het aantal wachtenden W1 en W2 in de twee wegvakken wordt voorgesteld in grafiekvorm, en wel zodanig dat het cijfer 1 de beeldpunten geeft van W1, en het cijfer 2 van W2.

Opm.: Wanneer de 1 en de 2 over elkaar zouden vallen wordt 1.2 afgedrukt, waarbij de . het beeldpunt weergeeft.

Het oppervlak tussen de W1 en W2 beeldpunten en de (vertikale) tijdas is gelijk aan het aantal 'STOPS'.

Er is een goede verkeersdoorstroming wanneer het aantal voertuigen dat per groen-rood fase vertrekt altijd groter is dan het aantal stilstaande voertuigen op het groenbegintijdstip.

In het geval dat er minder voertuigen per cyclus kunnen vertrekken dan er tijdens het groenbegintijdstip staan te wachten, kan het verkeer niet verwerkt worden en vindt (bij gelijkblijvend verkeersaanbod) accumulatie plaats van voertuigen voor de stopstreep. Dit zal zich uiten in een stijgend verloop van W_1 of W_2 , afgezien van de ruiseffecten.

In tweede instantie wordt per wegvak de aankomstverdeling over de cyclustijd afgebeeld in de vorm van een histogram. Voor ieder tijdstip in de cyclus zijn alle aankomsten op de stopstreep gesommeerd, zoals deze volgen uit de aankomstentabel.

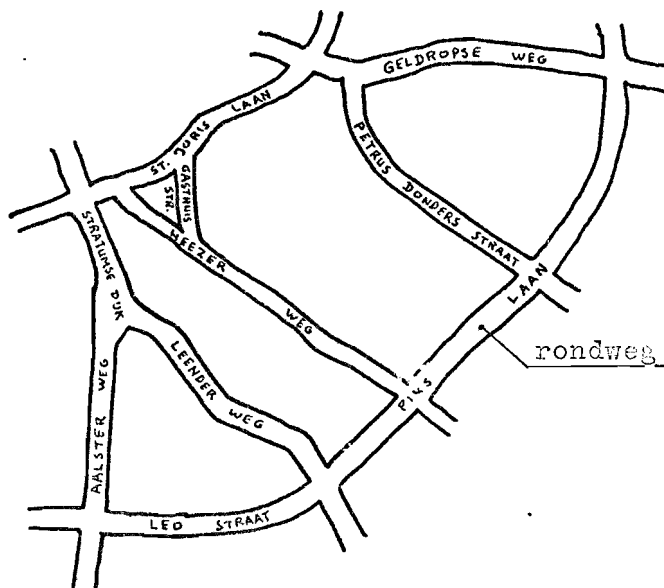
De aankomstverdeling van wegvakken, behorende tot de klasse ingaand zal vlak zijn, daar verkeer ingeloot wordt ten opzichte van een intensiteitsgetal dat constant is in de tijd.

In het algemeen zullen de aankomsten in verbindingswegvakken niet homogeen verdeeld zijn over de cyclus, daar het vertrek aan het begin van die wegvakken enigszins blokvormig periodiek is. Door de groen-offset zo te kiezen dat het maximale aantal aankomsten in de groen-fase komt te liggen, zal het aantal stops geminimaliseerd worden en ook de wachttijd stilstaand afnemen.

Door middel van de afkortingen G voor groen, en R voor rood, wordt langs de tijd-as van het histogram de verkeerslichten instelling aangegeven, gebruik makend van de effectieve groentijden.

8. Toepassing.

Het programma NETSIM is toegepast op een bestaand verkeersnetwerk in Eindhoven. Dank zij de medewerking van de afdeling "Ruimtelijke ordening en verkeer" van de Gemeente Eindhoven kon beschikt worden over gegevens van het netwerk: Rondweg(zuid)...St Jorislaan.



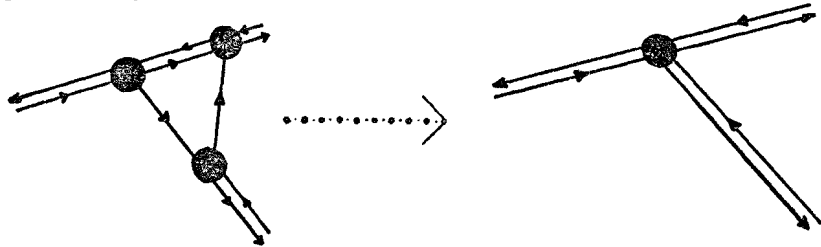
De verkeers-intensiteit gegevens zijn afgeleid uit telresultaten (voorjaar 1971). Voor ieder kruispunt zijn de tellingen gesommeerd per 15 minuten, zowel verdeeld naar oorsprong-bestemming als naar soort verkeer. De mogelijkheid bestond een karakteristiek verkeersbeeld voor de ochtendpiek en de avondpiek te onderscheiden, met voor beide pieken de dan geldende afbuigkansen. De ochtendpiek is verkregen uit gemiddelden over de periode 07.00-09.00 uur, en de avondpiek over de periode 16.00-18.00 uur.

De waarden van gemiddelde snelheid en spreiding in de verbindings wegvakken zijn geschat aan de hand van enkele summere gegevens. Op dit punt is er nog een duidelijke behoefte aan meetresultaten uit de praktijk.

De verkeerslichten instelling kon afgeleid worden uit de bestaande fase-diagrammen. Hierbij bleken de cyclustijden langs de rondweg 52, 62 of 70 seconden te zijn, en de cyclustijden op de overige kruispunten 50, 70, 80 of 90 seconden. Door de beperking van het programma NETSIM dat de cyclustijd in het gehele netwerk gelijk moet zijn, was het niet mogelijk de verkeerslichteninstelling zó te kiezen dat deze overeen stemde met de werkelijke instellingen tijdens de ochtend- en avondpiek.

Als compromis is voor het hele netwerk een cyclustijd van 70 seconden gekozen.

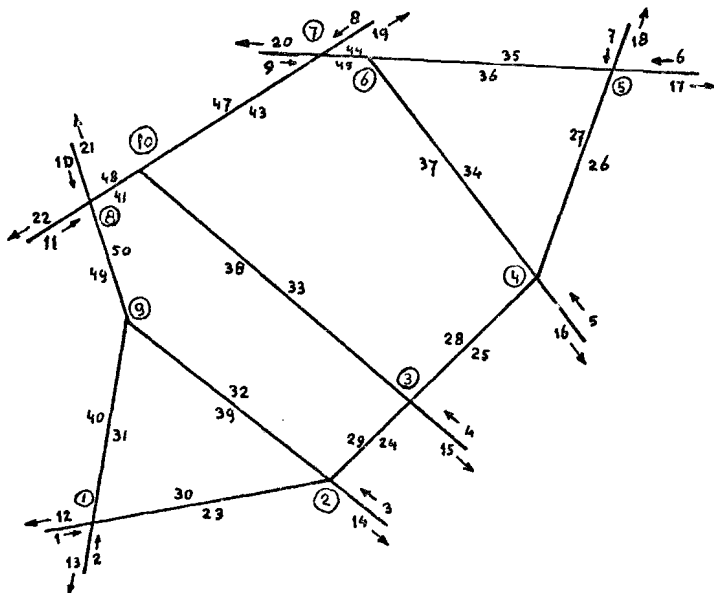
Als tweede probleem trad de afwezigheid op van verkeerslichten op de 3 driesprongen: Heezerweg- Gasthuisstraat- St.Jorislaan. In het programma is de geometrie van het netwerk gewijzigd door de drie kruispunten te vervangen door één driesprong, waarvan de verkeerslichten altijd groen zijn.



Deze oplossing is een benadering en mag alleen toegepast worden bij verkeers-intensiteiten van slechts enkele voertuigen per cyclustijd. Hinder en wachttijden op het onge-regelde kruispunt zijn dan in werkelijkheid klein, en kunnen in het simulatie model op het bewuste kruispunt niet optreden. "Altijd groen" moet gerealiseerd worden door het effectieve groenbegintijdstip aan het begin van de cyclustijd te kiezen, en het roodbegintijdstip buiten de cyclus. (Let op: de op te geven rood-begintijd moet groter zijn dan twee maal de cyclustijd in seconden.)

In dit geval zal tijdens executie van het programma de overgang groen → rood niet optreden, terwijl wel iedere cyclus op het groenbegintijdstip het aantal vertrokken voertuigen in de afgelopenperiode geregistreerd wordt.

Het netwerk bestaat na de aanpassing van de driesprongen Heezerweg- Gasthuisstraat- St.Jorislaan, uit 10 kruispunten en 48 wegvakken waarvan 11 ingaand, 11 uitgaand en 26 verbindend.



Er zijn na elkaar vier volledige simulaties S1, S2, S3 en S4 uitgevoerd, elk met een ander startgetal INT van de random-generator. De uitvoerige computer-output maakt geen deel uit van dit rapport, maar is separaat beschikbaar. Iedere simulatie beslaat 30 cycli (plus 3 inloop-cycli) van 70 seconden.

Om een indruk te krijgen van de stabiliteit van de simulatie worden hieronder in een aantal tabellen enkele karakteristieke grootheden gegeven voor de 8 verbindingswegvakken van de rondweg. (Avondpiek.)

wegvak	I N G E K O M E N				
	INTEN	S1	S2	S3	S4
23	713	756	758	719	750
24	691	717	703	672	694
25	678	692	683	659	695
26	646	672	646	624	701
27	574	569	546	554	544
28	571	587	548	544	532
29	539	540	516	513	494
30	532	562	538	509	509
totaal	4944	5059	4938	4794	4919

wegvak	WACHTTIJD STAANSTAND				VERTRAGING RIJDEND			
	S1	S2	S3	S4	S1	S2	S3	S4
23	7969	7293	5225	7777	126	100	125	91
24	1866	1644	1560	1683	86	69	78	90
25	2282	2329	2020	2314	79	61	65	50
26	4775	4130	4064	4918	71	53	59	100
27	2562	2367	2353	2505	49	50	42	42
28	5079	4718	4788	4608	81	61	106	79
29	6063	5452	7028	5687	90	88	83	77
30	5430	5745	4351	4369	33	26	23	23
totaal	56026	33678	31389	35941	615	508	581	552

De kolom INTEN geeft de verkeersverdeling zoals deze als stationnaire toestand resulteert uit het programma INTEN.

De stabiliteit van de avondpiek-simulatie is redelijk. In het algemeen kan gesteld worden dat de stabiliteit slechter wordt naarmate de capaciteit van een wegvak-uitmonding dichter benaderd wordt. Een oorzaak hiervoor is dat dan de invloed van mogelijke hinder zich uitstrekt over grotere aantallen voertuigen.

Opmerkelijk is dat de VERTRAGING RIJDEND zeer klein is vergeleken met de WACHTTIJD STILSTAAND. De grootte van VERTRAGING RIJDEND per voertuig dat een wegvak is ingekomen is klein genoeg om een te verwaarlozen vervorming van de rijtijdenverdeling te veroorzaken.

Opm. Als maximum waarde treedt alleen in de wegvakken 39 en 40 een VERTRAGING RIJDEND op van ongeveer één seconde per voertuig.

Als simulatie-resultaat zijn in het bijzonder de aankomst verdelingen interessant. Hierin komt de verkeersdoorstroming, speciaal op de sterk gekoppelde delen van de rondweg goed naar voren, waarbij een duidelijke overeenstemming met de praktijk geconstateerd kan worden.

Op de volgende pagina wordt de aankomstverdeling van wegvak 7 en wegvak 27 afgebeeld, opgetreden bij de avondpiek simulatie S2. Duidelijk is bij wegvak 7 de uniforme kansverdeling te herkennen, gestoord door ruis. De intensiteit van inkomend verkeer was 1200/uur of 700 per 30 cycli. Tijdens de simulatie zijn er 728 binnengekomen.

De aankomstverdeling van wegvak 27 heeft twee pieken; de belangrijkste bestaat uit verkeer uit wegvak 7, en heeft de top op fasetijd 19. De tweede piek bestaat uit afbuigend verkeer uit de wegvakken 6 en 36, en heeft de top op fasetijd 35.

lengte van wegvak27: 475 meter.
gemiddelde snelheid: 55 km/uur.
spreiding : 10 km/uur.

dit levert gemiddelde rijtijd: 31 sec.
heersende groen-offset : 34 sec.

FASE-
TIJD

AANKOMSTVERDELING WEGVAK 27.

R	1	8	*****
R	2	8	*****
R	3	9	*****
R	4	3	***
R	5	9	*****
R	6	4	****
R	7	6	*****
R	8	6	*****
R	9	5	*****
R	10	12	*****
R	11	10	*****
R	12	14	*****
R	13	17	*****
R	14	23	*****
G	15	32	*****
G	16	33	*****
G	17	38	*****
G	18	24	*****
G	19	36	*****
G	20	28	*****
G	21	38	*****
G	22	24	*****
G	23	31	*****
G	24	22	*****
G	25	19	*****
G	26	8	*****
G	27	14	*****
G	28	7	*****
G	29	3	***
G	30	3	***
G	31	3	***
G	32	8	*****
G	33	6	*****
R	34	13	*****
R	35	18	*****

542

FASE-
TIJD

AANKOMSTVERDELING WEGVAK 7.

G	1	19	*****
G	2	14	*****
G	3	22	*****
G	4	19	*****
G	5	18	*****
G	6	18	*****
G	7	24	*****
G	8	19	*****
G	9	30	*****
G	10	20	*****
G	11	18	*****
R	12	24	*****
R	13	22	*****
R	14	26	*****
R	15	24	*****
R	16	21	*****
R	17	26	*****
R	18	23	*****
R	19	23	*****
R	20	15	*****
R	21	20	*****
R	22	12	*****
R	23	17	*****
R	24	14	*****
R	25	19	*****
R	26	23	*****
R	27	23	*****
R	28	12	*****
R	29	21	*****
R	30	24	*****
R	31	19	*****
R	32	21	*****
G	33	18	*****
G	34	19	*****
G	35	22	*****

728

9. Resultaten.

Bij dit rapport is als bijlage de volledige programma-listing van NETSIM en INTEN toegevoegd.

De resultaten van de programma's, toegepast op een bestaand verkeersnetwerk van 50 wegvakken, 10 kruispunten, zijn separaat beschikbaar.

Door de grootte van het programma NETSIM en de extreem lange rekestijden op de IBM 360/30 installatie, is in een vroeg stadium besloten gebruik te maken van een IBM 360/75 systeem via het rekencentrum van de THE. Hierdoor stond een groot werkgeheugen (400 kbytes) en een F-compiler voor PL/I ter beschikking.

Om het hierboven genoemde netwerk gedurende 30 cycli te kunnen simuleren is in totaal 150 k geheugenruimte nodig. Per wegvak is de bijdrage hierin ongeveer 1 k.

Compileer- en Link-Edit stap samen vragen ongeveer 70 sec. Een volledige simulatie van 30 verkeerslicht-cycli (dit komt overeen met 35 minuten werkelijke tijd) kost, inclusief het componeren van een uitgebreide resultaten-output 60 tot 90 seconden, afhankelijk van de verkeersintensiteit.

10. Aanbevelingen.

In het huidige programma NETSIM wordt niet onderkend wanneer een wegvak "vol" is. Filelengtes hebben geen bovengrens. In het algemeen zal op een kruispunt waarvan één of meer bestemmingswegvakken vol zijn, het verkeersgedrag niet met voldoende zekerheid te voorspellen zijn om er een betrouwbaar model op te baseren. (Denk aan blokkering van het kruispunt door een file die staat te wachten om een vol wegvak in te kunnen rijden. De individuele mate van opvolging van de verkeersregels, (on)hoeffelijkheden en het harmonica-effect van lange files, maken dat vertaling van de werkelijkheid naar het simulatiemodel niet eenvoudig is.)

Momenteel moet het verloop van de filelengte tijdens de simulatie door de gebruiker kritisch beschouwd worden om in elk geval mogelijke volle wegvakken te kunnen opsporen. Men kan de moeilijkheid van volle wegvakken natuurlijk vermijden door bij het optreden ervan de simulatie te stoppen.

Concluderend kan gesteld worden dat het simulatiemodel zoals dit in het programma NETSIM gerealiseerd is, alleen zinvol toegepast kan worden wanneer geen extreme situaties optreden. Er moet, over het gehele netwerk gezien, een redelijke verkeersdoorstroming blijven bestaan; per wegvak is echter de optredende hinder, bepaald door geometrie en/of overig verkeer wel van invloed op het verkeersgedrag.

Het moet overwogen worden een goed simulatiemodel van het ongeregelde kruispunt toe te voegen. In de praktijk blijken er in de wat grotere netwerken vaak belangrijke wegvakken zonder verkeerslichten voor te komen.

Wanneer grotere netwerken gesimuleerd worden kan men niet ontkomen aan het invoeren van verschillende cyclustijden van de verkeerslichten. Ook zal dan in het netwerk, min of meer homogeen verdeeld, verkeer moeten kunnen ontstaan of verdwijnen, wat waarschijnlijk het beste gerealiseerd kan worden met behulp van een positieve of negatieve generatie-factor per wegvak. Bijvoorbeeld bestaat er, in het bijzonder op wegvakken langs woonwijken 's ochtends een duidelijk positief verkeersaanbod. Een maat voor dit verkeersaanbod kan zonder moeilijkheden uit de bestaande verkeersstellingen afgeleid worden.

11. Literatuur.

- [1] ANWB. Capaciteit van wegen en kruispunten.
Verkeers-memorandum No 2.
Den Haag, 1964.
- [2] P.J.Jeuken. Starre regeling van verkeer.
Afstudeerverslag THE No ECB-420
Eindhoven 1969.
- [3] E.Engel. Grundlagen der Strassenverkehrstechnik.
Sammlung Göschen, Band 1198
Walter de Gruyter & Co. Berlin 1961.
- [4] R.W.Hamming. Numerical Methods for Scientists and
Engineers. (pag. 34 en 389)
Mc Graw-Hill. N.Y. 1962.
- [5] D.I.Robertson. TRANSYT: A Traffic Network Study Tool.
Road Research Laboratory.
Crowthorne, Berkshire 1969.

SIMULATIE VAN EEN
MET VERKEERSLICHTEN
BEVEILIGD NETWERK.

(Bijlage)

H. NIJMAN.

Bijlage bij het gelijknamige rapport van het
Onderzoeksinstituut voor de Veiligheid van het
Verkeer (OVV) uitgevoerd gedurende de periode
15 september 1971 tot 15 juni 1972.

Inhoudsopgave.

I	PL/I programma NETSIM.	2
II	Assembler procedure RANDOM.	25
III	PL/I programma INTEN.	26

PROCEDURE OPTIONS (MAIN);

NETSIM: PROCEDURE OPTIONS (MAIN);

```

DECLARE (PRE,          /* AANTAL INLOOP-CYCLI */
        EX,           /* AANTAL VOLLEDEGE SIMULATIE-RUNS */
        KR,          /* AANTAL KRUISPUNTEN BINNEN HET NETWERK */
        AANTAL,      /* HOOGSTE VOLGNUMMER VAN DE WEGVAKKEN */
        CSEC,       /* TIJDSDUUR VAN EEN CYCLUS IN SECONDEN */
        CTAU,       /* AANTAL ACTIETIJDSTIPPEN PER CYCLUS */
        CSIM,       /* AANTAL CYCLI PER RUN NA HET INLOPEN */
        POS)       /* AANTAL POSITIES VAN DE AANKOMSTENTABEL */
        BINARY FIXED(15);
GET EDIT(PRE,EX,KR,AANTAL,CSEC,CTAU,CSIM,POS)(X(48),8(F(4)));
PUT EDIT(' INHOUD DATAKAARTEN.', 'BLAD 1')(A,X(92),A);PUT SKIP(3);
PUT EDIT(PRE,EX,KR,AANTAL,CSEC,CTAU,CSIM,POS)(X(48),8(F(4)));
BEGIN;
DECLARE (BUF,G,I,IA,IQ,IR,IV,J,JW,K,NK,NP,WG,WK,WV, /* HULPVAR. */
        NGG,NG,PRC,PRT,PRG,PRS, /* PRINTVAR. NETWERKSTATUS */
        RICHTING,RICHT1,RICHT2, /* GEKOZEN RICHTING */
        GROEN,ROOD, /* KLEUR VERKEERSLICHTEN */
        WEG,WEGA,WEGB,BES,TEG, /* NUMMERS VAN WEGVAKKEN */
        BLZ,REGEL, /* TELLERS PRINT-OUTPUT */
        TACT,TACT1,TACT2,TACT3, /* ACTUELE EN OPVOLGENDE KOLOMMEN
        /* VAN DE AANKOMSTENTABEL */
        AFW,UITG,ING,VERB, /* MOGELIJKE KLASSEN VAN WEGVAKKEN */
        CYCLUS,TFASE, /* TIJDSBEPALING IN DE SIMULATIERUN */
        VAN,NAAR, /* OORSPRONG EN BESTEMMINGSKRUISPUNT */
        MAXR,MAXL, /* MAXIMUM INHOUD OPSTELVAKKEN (DATA-IN) */
        PLAATS, /* GEWENSTE POSITIE IN DE AANKOMSTENTABEL */
        WAGENS, /* AANTAL VOERTUIGEN DAT DE STOPSTREEP BEREIKT */
        WEER, /* TELLER DIE AANGEEFT HOE VAAK EEN BEPAALD
        /* PROCES UITGEVOERD MOET WORDEN */
        (VORM, /* VORM VAN DE WEGVAK-UITMONDING */
        KLASSE, /* KLASSE VAN HET WEGVAK */
        RIJSTROKEN, /* AANTAL RIJSTROKEN VAN WEGVAK */
        AANKOMST1,AANKOMST2, /* AANKOMSTVAK MAX INHOUD: 1 AUTO */
        VAK21,VAK23, /* INHOUD OPSTELVAKKEN RECHTDOOR
        /* OP RECHTER EN LINKER RIJSTROOK */
        FILE1,FILE2, /* INHOUD FILES */
        AFSLAAND1,AFSLAAND2, /* AFSLAANDE AUTO'S IN EEN FILE */
        MAXTUSSENFILE, /* MAX INHOUD VAN DE TUSSENFILE */
        MAXBUFFER, /* MAXIMUM INHOUD VAN DE BUFFER */
        BUFFER, /* INHOUD VAN DE BUFFER */
        LENGTE, /* LENGTE VAN WEGVAK IN METERS */
        SNELHEID, /* GEMIDDELDE SNELHEID IN KM/H */
        SPREIDING, /* SPREIDING IN SNELHEID IN KM/H */
        WACHTEND, /* AANTAL WACHTENDE AUTO'S VOOR STOPSTREEP */
        RIJDEND, /* AANTAL RIJDENDE AUTO'S IN EEN WEGVAK */
        VERTREK, /* AANTAL AUTO'S KLAAR VOOR VERTREK */

```

PROCEDURE OPTIONS (MAIN);

```

    VERTREKCYC, /* UIT WEGVAK VERTROKKEN SINDS GROENBEGIN */
    VERTREKSOM, /* TOTAAL UIT WEGVAK VERTROKKEN */
    TEGENLIGGER) /* TEGEMOETKOMEND WEGVAK OP KRUISPUNT */
    /* BOVENSTAANDE ARRAY'S HEBBEN. */
    (AANTAL), /* 'AANTAL' ELEMENTEN */
    (VAK, /* INHOUD VAN DE OPSTELVAKKEN */
    MAXVAK, /* MAXIMUM INHOUD VAN DE OPSTELVAKKEN */
    LICHT, /* KLEUR VAN HET VERKEERSLICHT */
    ROODBEGIN, /* ACTIETIJDSTIP DAT LICHT ROOD WORDT */
    GROENBEGIN, /* ACTIETIJDSTIP DAT LICHT GROEN WORDT */
    BESTEMMING) /* BESTEMMINGSWEGVAK BIJ KEUZE L,S,R */
    /* BOVENSTAANDE ARRAY'S HEBBEN */
    (AANTAL,3), /* 'AANTAL' * 3 ELEMENTEN */
    BREEDTE /* AANTAL AUTO'S DAT GELIJKTIJDIG */
    /* NAAST ELKAAR KAN VERTREKKEN UIT */
    /* DE OPSTELVAKKEN EN DE BUFFER */
    (AANTAL,4), /* 'AANTAL' * 4 ELEMENTEN */
    (GROENFILE, /* WACHTENDEN WANNEER HET */
    /* VERKEERSLICHT GROEN WORDT */
    PERIODEVERTREK) /* VERTROKKEN IN AFGELOPEN */
    /* GROEN-ROOD PERIODE */
    (AANTAL,CSIM), /* 'AANTAL'* CSIM ELEMENTEN */
    B /* AANKOMSTENTABEL */
    /* BESTAANDE UIT 'AANTAL' */
    (AANTAL,POS), /* RIJEN EN POS KOLOMMEN */
    AANK_VERDELING /* AANKOMST VERDELING */
    /* IN DE CYCLUS */
    (AANTAL,CTAU)) /* 'AANTAL'* CTAU ELEMENTEN */

    BINARY FIXED(15);
    DECLARE (IWTOTAAL,IWTOT,INT,IS,IT,IU,IW,IX,IY,IZ, /* HULPVARIABLEN */
    (STOPS, /*AANTAL STOPS GESOMMEERD OP GROENBEGINTIJD */
    WACHTTIJD, /* WACHTTIJD VAN ALLE STILSTAANDE AUTO'S */
    VERTRAGING) /* VERTRAGING OPGELOPEN IN HET WEGVAK */
    /* DOORDAT EEN GEWENSTE POSITIE BEZET WAS */
    (AANTAL)) /* 'AANTAL' ELEMENTEN */

    BINARY FIXED(31);
    DECLARE (A,SNELH,RIJTIJD, /* HULPVARIABLEN */
    TAU, /* TIJDSDUUR TUSSEN TWEE ACTIETIJDSTIPPEN */
    (RECHTSAF, /* FRACTIE RECHTSAFSLAAND VERKEER */
    NIETLINKSAF, /* FRACTIE NIET LINKSAFSLAAND VERKEER */
    INTENS) /* INTENSITEIT INKOMEND VERKEER IN VOER- */
    /* TUIGEN PER ACTIETIJDSTIP EN RIJSTROOK */
    (AANTAL)) /* 'AANTAL' ELEMENTEN */

    BINARY FLOAT;
    DECLARE (NORM, /* HULPVARIABLE BIJ RIJTIJDBEPALING */
    REAL) /* HULPVARIABLE IN RANDOM PROCEDURE */
    BINARY FLOAT(21);

```

PROCEDURE OPTIONS (MAIN);

```

/*  INITIALISATIE ALGEMEEN.                                     */
    MAXVAK          =0;  BLZ   =1;  REGEL=4;  KLASSE   =0;
    MAXBUFFER       =0;  BUF   =4;  PRC   =0;  INTENS  =0;
    MAXTUSSENFILE   =0;  AFW   =0;  PRG   =0;  BREEDTE  =1;
    LENGTE          =0;  UITG  =1;  PRS   =0;  GROEN   =0;
    SNELHEID        =0;  ING   =2;  PRT   =0;  ROOD    =1;
    SPREIDING       =0;  VERB  =3;  NORM  =0;  INT     =1;

/*  DE NETWORK-DATA WORDEN INGELEZEN.                           */
A1:  DO I=0 TO AANTAL;
      GET EDIT (WEG)(F(3));
      REGEL=REGEL+1; IF REGEL>60
          THEN DO; BLZ=BLZ+1;  REGEL=2;
                PUT PAGE EDIT('BLAD',BLZ)(X(112),A,
                F(4));  PUT SKIP;
          END;
                PUT SKIP EDIT (WEG)(F(3));
      IF WEG=0 THEN DO; GET EDIT(K)(X(76),F(1));
                PUT EDIT(K)(X(76),F(1));          GOTO AAA;
      END;
      GET EDIT (KLASSE(WEG),RIJSTROKEN(WEG))(2(F(2)));
      PUT EDIT (KLASSE(WEG),RIJSTROKEN(WEG))(2(F(2)));
      IF KLASSE(WEG)=1 THEN DO; GET EDIT(VAN,NAAR)(X(65),2(F(4)));
                PUT EDIT(VAN,NAAR)(X(65),2(F(4)));
                GOTO ENDA1;
      END;
      GET EDIT (VORM(WEG),
      BESTEMMING(WEG,3),BESTEMMING(WEG,2),BESTEMMING(WEG,1),
      TEGENLIGGER(WEG),RECHTSAF(WEG),NIETLINKSAF(WEG),
      GROENBEGIN(WEG,3),GROENBEGIN(WEG,2),GROENBEGIN(WEG,1),
      ROODBEGIN (WEG,3),ROODBEGIN (WEG,2),ROODBEGIN (WEG,1),
      MAXBUFFER(WEG))(F(2),4(F(4)),9(F(3)));
      PUT EDIT (VORM(WEG),
      BESTEMMING(WEG,3),BESTEMMING(WEG,2),BESTEMMING(WEG,1),
      TEGENLIGGER(WEG),RECHTSAF(WEG),NIETLINKSAF(WEG),
      GROENBEGIN(WEG,3),GROENBEGIN(WEG,2),GROENBEGIN(WEG,1),
      ROODBEGIN (WEG,3),ROODBEGIN (WEG,2),ROODBEGIN (WEG,1),
      MAXBUFFER(WEG))(F(2),4(F(4)),9(F(3)));
      IF RIJSTROKEN(WEG)>2
      THEN DO J=3,2,1,BUF;
                GET EDIT (BREEDTE(WEG,J))(F(2));
                PUT EDIT (BREEDTE(WEG,J))(F(2));
      END;
      ELSE DO; GET EDIT(MAXL,MAXR,BREEDTE(WEG,BUF))(2(F(3)),F(2));
                PUT EDIT(MAXL,MAXR,BREEDTE(WEG,BUF))(2(F(3)),F(2));
      END;
      IF KLASSE(WEG)=VERB
      THEN DO; GET EDIT (LENGTE(WEG),SNELHEID(WEG),SPREIDING(WEG),

```

PROCEDURE OPTIONS (MAIN);

```

VAN,NAAR)(F(5),2(F(3)),X(1),2(F(4)));
PUT EDIT (LENGTE(WEG),SNELHEID(WEG),SPREIDING(WEG),
VAN,NAAR)(F(5),2(F(3)),X(1),2(F(4)));
END;
ELSE DO; GET EDIT (INTENS(WEG),VAN,NAAR)(F(5),X(7),2(F(4)));
PUT EDIT (INTENS(WEG),VAN,NAAR)(F(5),X(7),2(F(4)));
END;
/* UITGAANDE VAN DE GEOMETRIE-GEGEVENS VAN DE WEGEN */
/* WORDEN DE KARAKTERISTIEKE GROOTHEDEN */
/* MAXVAK EN MAXTUSSENFILE BEREKEND. */
IF VORM(WEG)=5 THEN GOTO ENDA1;
MAXVAK(WEG,1)=MAXR+1;
MAXVAK(WEG,3)=MAXL+1;
MAXTUSSENFILE(WEG)=MAXR-MAXL;
IF VORM(WEG)=4 THEN DO; BREEDTE(WEG,2)=2; GOTO ENDA1;
END;
IF VORM(WEG)=2 THEN DO;
MAXVAK(WEG,2)=MAXVAK(WEG,1);
GOTO ENDA1;
END;
IF VORM(WEG)=3 THEN DO;
MAXVAK(WEG,2)=MAXVAK(WEG,1);
MAXTUSSENFILE(WEG)=-MAXTUSSENFILE(WEG)-1;
GOTO ENDA1;
END;
MAXTUSSENFILE(WEG)=MAXTUSSENFILE(WEG)-1;
MAXVAK(WEG,2)=MAXVAK(WEG,3);
ENDA1: END;
/* DATA-KAART INLEZEN WELKE AANGEEFT WELKE ACTIETIJDSTIPPEN ALLE */
/* TUSSENRESULTATEN UITGEPRINT MOETEN WORDEN. */
/* NG= AANTAL MALEN PER SIMULATIE DAT TUSSENRESULTATEN UITGEPRINT */
/* MOETEN WORDEN. */
/* PRC= CYCLUS TIJDENS WELKE DE EERSTE TUSSENRESULTATEN UITGEPRINT */
/* WORDEN. */
/* PRT= FASETIJD IDEM. */
/* PRG= AANTAL OPVOLGENDE ACTIETIJDSTIPPEN WAARVAN DE TUSSENRESUL- */
/* TATEN GEPRINT WORDEN. */
/* PRS= AANTAL DAAROP VOLGENDE ACTIETIJDSTIPPEN DAT GEEN */
/* TUSSENRESULTATEN GEPRINT WORDEN. */
/* PRG EN PRS WORDEN CYCLISCH DOORLOPEN TOT NG BEREIKT IS. */
AAA: GET EDIT(NG)(F(4)); NGG=NG;
REGEL=REGEL+1; IF REGEL>55
THEN DO; BLZ=BLZ+1; REGEL=2;
PUT PAGE EDIT('BLAD',BLZ)(X(112),A,
F(4));
END;
IF NG<1 THEN DO; GET EDIT(K)(X(75),F(1));

```


PROCEDURE OPTIONS (MAIN);

```

                PUT SKIP      EDIT(NG,K)(F(4),X(75),F(1));
            END;
        ELSE DO; GET EDIT(PRC,PRT,PRG,PRS,K)
                (4(F(4)),X(59),F(1));
                PUT SKIP      EDIT(NG,PRC,PRT,PRG,PRS,K)
                (5(F(4)),X(59),F(1));
            END;
    PUT SKIP(2) EDIT(' DE DATAKAARTEN WELKE DE WEGEN AANGEVEN WAARVAN UITGE
    BREIDE SIMULATIE-', 'RESULTATEN WORDEN GEVRAAGD ZIJN HIER NIET AFGEDRUKT
    .')(A,SKIP,X(1),A);
    /*      UITPRINTEN NETWERKGEGEVENS EN HET                                */
    /*      NORMEREN VAN INTENSITEITEN, AFBUIGPERCENTAGES                    */
    /*      EN GROEN- ROODTIJDSTIPPEN.                                       */
    TAU=CSEC/CTAU;
    BLZ=BLZ+1; REGEL=1; PUT PAGE EDIT ('BLAD',BLZ)(X(112),A,F(4));
    PUT EDIT(' DE CYCLUSTIJD BEDRAAGT ',CSEC,' SECONDEN, VERDEELD
    IN ',CTAU,' ACTIETIJDSTIPPEN.', ' DE TIJD TUSSEN TWEE ACTIETIJDSTIPPEN
    IS ',TAU,' SECONDEN.', ' DE BETEKENIS VAN DE IN ONDERSTAANDE TABEL G
    EBRUIKTE BENAMINGEN IS DE VOLGENDE:', 'SEC      TIJDSTIP IN SECON
    DEN.', 'ACTIE      TIJD IN ACTIETIJDSTIPPEN.', 'NR      VOLG
    NUMMER VAN DE WEG.', 'RIJ      AANTAL RIJSTROKEN.',
    'VORM      VORMGETAL BEPAALD DOOR DE GEOMETRIE VAN DE WEG EN DE O
    PSTELVAKKEN.', 'BES3      BESTEMMINGSWEG LINKSAF.', 'BES2
    BESTEMMINGSWEG RECHTDOOR.', 'BES1      BESTEMMINGSWEG RECHTSAF.'
    , 'TEGL      DE TEGEMOETKOMENDE WEG.', '%-L      PERCENTAGE
    NIET-LINKSAFSLAAND VERKEER.', '%R      PERCENTAGE RECHTSAFSLAAND
    VERKEER.', 'MAX-VAKKEN      MAXIMUM INHOUD VAN DE OPSTELVAKKEN.',
    'MF      MAXIMUM INHOUD VAN DE TUSSENFILE.', 'MBUF      MA
    XIMUM INHOUD VAN DE BUFFER.', 'GROENBEGINTIJD      GROENTIJDSTIPPEN VOOR DE
    OPSTELVAKKEN.', 'ROODBEGINTIJD      ROODTIJDSTIPPEN VOOR DE OPSTELVAKKEN.',
    'BREEDTE      BREEDTE VAN DE OPSTELVAKKEN.', 'INT      INTENSI
    TEIT VAN HET INKOMEND VERKEER PER RIJSTROOK PER ACTIETIJDSTIP.',
    'LANG      LENGTE VAN DE WEG IN METERS.', 'SNEL      GEMIDDE
    LDE RIJSNELHEID IN KM/H.', 'SPR      SNELHEIDSSPREIDING IN KM/H.'
    , 'ING      INGAANDE WEG.', 'VERB      VERBINDINGSWEG.',
    'UITG      UITGAANDE WEG.')
    (2(A,F(3)),A,SKIP,A,F(5,3),A,SKIP(3),A,SKIP(2),24(A,SKIP));
    PUT SKIP(2); REGEL=REGEL+32; CALL KOPS(BLZ,REGEL);
    A2: DO I=1 TO AANTAL;
        IF KLASSE(I)=AFW THEN GOTO ENDA2;
        IF REGEL>58 THEN CALL KOPS(BLZ,REGEL); REGEL=REGEL+2;
        PUT SKIP EDIT (I,RIJSTROKEN(I))(F(3),F(4));
        IF KLASSE(I)=UITG THEN DO; PUT EDIT(' UITG')(A); PUT SKIP;
            GOTO ENDA2;
        -ND;
        PUT EDIT (VORM(I),BESTEMMING(I,3),
        BESTEMMING(I,2),BESTEMMING(I,1),TEGENLIGGER(I),

```

PROCEDURE OPTIONS (MAIN);

```

        NIETLINKSAF(I),RECHTSAF(I))(5(F(5)),X(1),2(F(4)));
    IF VORM(I)=5
    THEN PUT EDIT (MAXBUFFER(I))(X(16),F(4));
    ELSE DO; IF VORM(I)=4 THEN PUT EDIT(MAXVAK(I,3),MAXVAK(I,1),
        MAXTUSSENFILE(I),MAXBUFFER(I))
        (F(4),X(4),3(F(4)));
        ELSE PUT EDIT(MAXVAK(I,3),MAXVAK(I,2),
        MAXVAK(I,1),MAXTUSSENFILE(I),
        MAXBUFFER(I))(5(F(4)));

        END;
    PUT EDIT(
        GROENBEGIN(I,3),GROENBEGIN(I,2),GROENBEGIN(I,1),
        ROODBEGIN(I,3),ROODBEGIN(I,2),ROODBEGIN(I,1),
        BREEDTE(I,3),BREEDTE(I,2),BREEDTE(I,1),BREEDTE(I,BUF))
        (6(F(5)),X(1),4(F(2)));
    IF KLASSE(I)=VERB
    THEN PUT EDIT (' VERB ',LENGTE(I),SNELHEID(I),SPREIDING(I))
        (A(5),3(F(5)));
    ELSE DO; INTENS(I)=(INTENS(I)*TAU)/(RIJSTROKEN(I)*3600);
        PUT EDIT (INTENS(I),' ING')(F(5,2),A(5));

        END;
        RECHTSAF(I)=RECHTSAF(I)/100;
        NIETLINKSAF(I)=NIETLINKSAF(I)/100;
/*  OMREKENING GROEN- ROODTIJDSTIPPEN VAN                                */
/*  SECONDEN NAAR ACTIETIJDSTIPPEN.                                     */
        DO J=1 TO 3;
            A=(4+GROENBEGIN(I,J))/TAU+.5; IA=A;
            IF IA>CTAU THEN IA=IA-CTAU;
            GROENBEGIN(I,J)=IA;
            A=ROODBEGIN(I,J)/TAU+0.5; IA=A;
            IF IA>CTAU THEN IA=IA-CTAU;
            ROODBEGIN(I,J)=IA;
/*  INITIALISATIE VERKEERSLICHTEN                                       */
            IF GROENBEGIN(I,J) > IA THEN LICHT(I,J)=GROEN;
            ELSE LICHT(I,J)=ROOD;

        END;
        PUT SKIP EDIT(GROENBEGIN(I,3),GROENBEGIN(I,2),GROENBEGIN
        (I,1),ROODBEGIN(I,3),ROODBEGIN(I,2),ROODBEGIN(I,1))
        (X(61),6(F(5)));
    ENDA2: END;
/*  OP DEZE PLAATS BEGINT IEDERE NIEUWE SIMULATIE.                       */
RET:  VERTREK          =0;  AANKOMST1 =0;  RIJDEND      =0;  VAK          =0;
      VERTREKCYC       =0;  AANKOMST2 =0;  STOPS        =0;  VAK21       =0;
      TACT              =0;  AFSLAAND1 =0;  B            =0;G=0;  WAK23      =0;
      TACT1            =1;  AFSLAAND2 =0;  CYCLUS=1-PRE;  FILE1       =0;
      TACT2            =2;  WACHTIEND =0;  NG           =NGG;  FILE2      =0;
      TACT3            =3;  TFASE     =0;  REGEL      = 60;  BUFFER     =0;

```

PROCEDURE OPTIONS (MAIN);

```
/* VERKEERSVERWERKING. */
/* ALLEREERST WORDT DE KLOK WELKE HET */
/* VOLGENDE ACTIETIJDSTIP AANGEEFT GELIJKGEZET. */
SIM:   TFASE=TFASE+1;
      IF TFASE>CTAU THEN DO; TFASE=1; CYCLUS=CYCLUS+1;
      IF CYCLUS>CSIM THEN GOTO PRINT_ALG;
      END;
      IF CYCLUS=1 & TFASE=1
/* VANAF DIT MOMENT WORDEN DE TELGEGEVENS VERWERKT. */
      THEN DO; AANK_VERDELING =0; VERTREKSOM=0;
      WACHTTIJD =0; VERTRAGING=0;
      DO I=1 TO AANTAL;
      IF KLASSE(I)=UITG THEN RIJDEND(I)=0;
      END; END;
/* DE TIJD-WIJZERS WELKE DE ACTUELE EN EERSTVOLGENDE */
/* KOLOMMEN VAN DE AANKOMSTENTABEL AANGEVEN */
/* WORDEN BIJGESTELD. */
      TACT=TACT1; TACT1=TACT2; TACT2=TACT3;
      TACT3=TACT3+1; IF TACT3>POS THEN TACT3=1;
/* IN DE NU VOLGENDE DO-LOOP WORDEN: */
/* A) DE VERKEERSLICHTEN INGESTELD AAN DE */
/* HAND VAN DE OPGEGEVEN GROEN- ROOD TIJDSTIPPEN. */
/* B) HET AANTAL VOERTUIGEN BEPAALD DAT DE */
/* STOPSTREEP ZAL BEREIKEN. */
/* C) DE VERVOLGRICHTINGEN BEPAALD EN WORDEN */
/* DE VOERTUIGEN GEPLAATST IN DE AANKOMSTVAKKEN, FILES ED. */
B1:   DO I=1 TO AANTAL;
      IF KLASSE(I)<2 THEN GOTO ENDB1;
      DO J=1 TO 3;
      IF TFASE=GROENBEGIN(I,J)
      THEN DO; LICHT(I,J)=GROEN;
      IF J=2 THEN
      DO; IF CYCLUS >0 THEN
      DO; GROENFILE(1,CYCLUS)=WACHTEND(I);
      STOPS(I)=STOPS(I)+WACHTEND(I);
      PERIODEVERTREK(I,CYCLUS)=VERTREKCYC(I);
      END;
      VERTREKCYC(I)=0;
      END; END;
      ELSE IF TFASE=ROODBEGIN(I,J) THEN LICHT(I,J)=ROOD;
      END;
      IF KLASSE(I)=ING
      THEN DO; WEER=RIJSTROKEN(I); WAGENS=0;
      DO J=1 TO WEER;
      CALL RANDOM(INT,REAL);
      IF REAL < INTENS(I) THEN WAGENS=WAGENS+1;
      END;
      END;
```

```

        B(I,TACT3)=WAGENS;
        RIJDEND(I)=RIJDEND(I)+WAGENS;
    END;
    WAGENS=B(I,TACT);    B(I,TACT)=0;
    AANK_VERDELING(I,TFASE)=AANK_VERDELING(I,TFASE)+WAGENS;
    RIJDEND(I)=RIJDEND(I)-WAGENS;
    IF VORM(I)<4 THEN GOTO B1G;
    IF VORM(I)=5
/*  DE WEG HEEFT MEER DAN TWEE RIJSTROKEN.                                */
    THEN DO; DO WHILE (WAGENS > 0);
        CALL RICHT(RECHTSAF(I),NIETLINKSAF(I),RICHTING);
        VAK(I,RICHTING)=VAK(I,RICHTING)+1;
        WAGENS=WAGENS-1;
    END;
    GOTO ENDB1;
    END;
/*  DE WEG HEEFT TWEE RIJSTROKEN.                                        */
    IF WAGENS=0 THEN GOTO B1F;
    RICHT2=0;    CALL RICHT(RECHTSAF(I),NIETLINKSAF(I),RICHT1);
    IF WAGENS=2
    THEN DO;    CALL RICHT(RECHTSAF(I),NIETLINKSAF(I),RICHT2);
        IF RICHT1=2
/*  INDIEN NOODZAKELIJK WORDT DE INHOUD                                */
/*  VAN RICHT1 EN RICHT2 OMGEWISSELD.                                  */
        THEN DO; RICHT1=RICHT2; RICHT2=2;
    END;    END;
/*  NIEUW AANGEKOMEN VOERTUIGEN WORDEN EERST IN DE FILES GEPLAATST. */
    IF RICHT1=2
    THEN DO; IF RICHT2=2
        THEN DO; FILE1(I)=FILE1(I)+1;
            FILE2(I)=FILE2(I)+1;
        END;
        ELSE DO; IF FILE2(I) < FILE1(I)+MAXTUSSENFILE(I)
            THEN FILE2(I)=FILE2(I)+1;
            ELSE FILE1(I)=FILE1(I)+1;
        END;
    GOTO B1F;
    END;
/*  NU GELDT: RICHT1=1 OF 3 ZODAT HET BETREFFENDE VOERTUIG KAN        */
/*  AANSLUITEN BIJ DE RECHTER- RESP. LINKERFILE.                      */
/*  WANNEER RICHT2=2 WORDT HET VOERTUIG IN DE ANDERE FILE GEPLAATST.*/
    IF RICHT1=1
    THEN DO: FILE1(I)=FILE1(I)+1;
        AFSLAAND1(I)=AFSLAAND1(I)+1;
        IF RICHT2=0 THEN GOTO B1F;
        IF RICHT2=1
        THEN DO; FILE1(I)=FILE1(I)+1;

```

PROCEDURE OPTIONS (MAIN);

```

        AFSLAAND1(I)=AFSLAAND1(I)+1;
        GOTO B1F;
    END;
    FILE2(I)=FILE2(I)+1;
    IF RICHT2=3 THEN AFSLAAND2(I)=AFSLAAND2(I)+1;
END;
ELSE DO; FILE2(I)=FILE2(I)+1;
AFSLAAND2(I)=AFSLAAND2(I)+1;
IF RICHT2=0 THEN GOTO B1F;
IF RICHT2=3
THEN DO; FILE2(I)=FILE2(I)+1;
AFSLAAND2(I)=AFSLAAND2(I)+1;
GOTO B1F;
END;
FILE1(I)=FILE1(I)+1;
IF RICHT2=1 THEN AFSLAAND1(I)=AFSLAAND1(I)+1;
END;
B1F: IF AANKOMST1(I)=0 & FILE1(I)≠0
/* ER STAAT RECHTS NOG EEN FILE TERWIJL HET AANKOMSTVAK LEEG IS. */
THEN DO; CALL VUL(FILE1(I),AFSLAAND1(I),RICHTING);
IF RICHTING=2 THEN AANKOMST1(I)=2;
ELSE DO; AANKOMST1(I)=1;
AFSLAAND1(I)=AFSLAAND1(I)-1;
END;
FILE1(I)=FILE1(I)-1;
END;
IF AANKOMST2(I)=0 & FILE2(I)≠0
/* ER STAAT LINKS NOG EEN FILE TERWIJL HET AANKOMSTVAK LEEG IS. */
THEN DO; CALL VUL(FILE2(I),AFSLAAND2(I),RICHTING);
IF RICHTING=2 THEN AANKOMST2(I)=2;
ELSE DO; AANKOMST2(I)=3;
AFSLAAND2(I)=AFSLAAND2(I)-1;
END;
FILE2(I)=FILE2(I)-1;
END;
/* PROBEER DE INHOUD VAN DE AANKOMSTVAKKEN */
/* IN DE OPSTELVAKKEN TE PLAATSEN. */
IF AANKOMST1(I)≠0
THEN DO; IF VAK(I,1) < MAXVAK(I,1) & VAK21(I) < MAXVAK(I,1)
THEN DO; IF AANKOMST1(I)=1
THEN VAK(I,1)=VAK(I,1)+1;
ELSE DO; VAK(I,2)=VAK(I,2)+1;
VAK21(I)=VAK21(I)+1;
END;
AANKOMST1(I)=0;
END;
END;
IF AANKOMST2(I)≠0

```

PROCEDURE OPTIONS (MAIN);

```

THEN DO; IF VAK(I,3) < MAXVAK(I,3) & VAK23(I) < MAXVAK(I,3)
      THEN DO; IF AANKOMST2(I)=3
            THEN VAK(I,3)=VAK(I,3)+1;
            ELSE DO; VAK(I,2)=VAK(I,2)+1;
                    VAK23(I)=VAK23(I)+1;
            END;
            AANKOMST2(I)=0;
      END;
END;
GOTO ENDB1;

```

B1G:

```

/* DE WEG HEEFT EEN RIJSTROOK. */
IF WAGENS=1
THEN DO; IF AANKOMST1(I)=0
      THEN CALL RICHT(RECHTSAF(I),NIETLINKSAF(I),
                    AANKOMST1(I));
      ELSE FILE1(I)=FILE1(I)+1;
      END;
ELSE DO; IF AANKOMST1(I)=0 & FILE1(I)=0
      THEN DO; IF VORM(I)=2 THEN GOTO ENDB1;
              GOTO B1J;
      END;
      IF AANKOMST1(I)=0
      THEN DO; FILE1(I)=FILE1(I)-1;
              CALL RICHT(RECHTSAF(I),NIETLINKSAF(I),
                    AANKOMST1(I));
      END;
      END;
      IF VORM(I)=2

```

```

/* OPNAME IN EEN DER OPSTELVAKKEN. */
THEN DO; IF VAK(I,1) < MAXVAK(I,1) &
      VAK(I,2) < MAXVAK(I,2) &
      VAK(I,3) < MAXVAK(I,3)
      THEN DO; VAK(I,AANKOMST1(I))=VAK(I,AANKOMST1(I))+1;
              AANKOMST1(I)=0;
      END;
      GOTO ENDB1;
      END;
      IF AANKOMST1(I)=VORM(I)

```

```

/* OPNAME IN HET LANGSTE OPSTELVAK. */
THEN DO; IF VAK(I,VORM(I)) < MAXVAK(I,VORM(I))
      & FILE2(I) < MAXTUSSENFILE(I)
      THEN DO; VAK(I,VORM(I))=VAK(I,VORM(I))+1;
              AANKOMST1(I)=0;
      END;
      END;

```

```

B1J:      IF FILE2(I)=0 & AANKOMST2(I)=0 THEN GOTO ENDB1;
          IF FILE2(I)>0 & AANKOMST2(I)=0

```

```

/* HET TWEEDE AANKOMSTVAK WORDT GEVULD */
/* REKENING HOUDEND MET DE FRACTIE */

```

```

/*  AFSLAAND VERKEER IN DE TUSSENFILE.                                     */
      THEN DO; CALL VUL(FILE2(I),AFSLAAND2(I),RICHTING);
              FILE2(I)=FILE2(I)-1;
              IF RICHTING=2
              THEN AANKOMST2(I)=2;
              ELSE DO; AANKOMST2(I)=4-VORM(I);
                      AFSLAAND2(I)=AFSLAAND2(I)-1;
              END;      END;      END;
      ELSE DO; IF AANKOMST2(I)=0
/*  HET VOERTUIG SLUIT ACHTER AAN BIJ DE EVENTUEEL                       */
/*  AANWEZIGE TUSSENFILE TERWIJL HET VOERTUIG DAT                       */
/*  VOORAAN STAAT DE FILE VERLAAT MET GELIJKE                           */
/*  BESTEMMING EN HET VAKJE AANKOMST2 OPVULT.                            */
      THEN DO; AANKOMST2(I)=AANKOMST1(I);
              AANKOMST1(I)=0;
              END;
      ELSE DO; IF FILE2(I) < MAXTUSSENFILE(I)
/*  OPNAME IN DE TUSSENFILE.                                             */
      THEN DO; FILE2(I)=FILE2(I)+1;
              IF AANKOMST1(I)≠2
              THEN AFSLAAND2(I)=AFSLAAND2(I)+1;
              AANKOMST1(I)=0;
              END;      END;      END;
      IF VAK(I,4-VORM(I)) < MAXVAK(I,2)
      & VAK(I,2) < MAXVAK(I,2)
/*  OPNAME IN HET KORTSTE OPSTELVAK.                                     */
      THEN DO; VAK(I,AANKOMST2(I))=VAK(I,AANKOMST2(I))+1;
              AANKOMST2(I)=0;
              END;
ENDB1:  END;
/*  IN DEZE DO-LOOP WORDEN DE KRUISPUNTEN                               */
/*  GEDEELTELIJK AFGEWERKT NL:                                          */
/*  A) DE BUFFERS VOOR LINKSAF WORDEN GEVULD.                            */
/*  B) INDIEN ER GEEN HINDER IS :  VOORLOPIG VERTREK UIT DE BUFFER.    */
/*  C) VOORLOPIG VERTREK RECHTSAF UIT VAK1.                              */
B2:    DO I=1 TO AANTAL;
      IF KLASSE(I)<2 THEN GOTO ENDB2;
      IF LICHT(I,3)=GROEN
      THEN DO; WEER=BREEDTE(I,3);
              DO WHILE (WEER >0 & BUFFER(I) < MAXBUFFER(I)
                      & VAK(I,3) > 0 );
              VAK(I,3)=VAK(I,3)-1;
              BUFFER(I)=BUFFER(I)+1;
              WEER=WEER-1;
              END;END;
      IF BUFFER(I) >0
      THEN DO; TEG=TEGENLIGGER(I);

```

PROCEDURE OPTIONS (MAIN);

```

IF TEG=0 THEN GOTO B2C;
IF LICHT(TEG,1)=ROOD &
  LICHT(TEG,2)=ROOD
THEN GOTO B2C;
IF LICHT(TEG,1)=GROEN
THEN DO; IF VAK(TEG,1) >0 THEN GOTO B2E;
  END;
IF LICHT(TEG,2)=GROEN
THEN DO; IF VAK(TEG,2) >0 THEN GOTO B2E;
  END;

```

```

/* TEST OP HINDER VAN TEGENLIIGERS. */
IF AANKOMST1(TEG)≠0 |
  AANKOMST2(TEG)≠0 |
  FILE1 (TEG)≠0 |
  FILE2 (TEG)≠0 |
  B (TEG,TACT1)≠0 |
  B (TEG,TACT2)≠0 |
  B (TEG,TACT3)≠0
THEN GOTO B2E;

```

```

B2C:
/* ER IS GEEN HINDER VOOR LINSAFSLAAND VERKEER. */
WEER=BREEDTE(I,BUF);
BES=BESTEMMING(I,3);
DO WHILE (WEER > 0 & BUFFER(I) > 0);
  VERTREKCYC(I)=VERTREKCYC(I)+1;
  VERTREKSOM(I)=VERTREKSOM(I)+1;
  VERTREK(BES)=VERTREK(BES)+1;
  BUFFER(I)=BUFFER(I)-1;
  WEER=WEER-1;
END;END;

```

```

B2E:
IF LICHT(I,1)=GROEN
THEN DO; WEER=BREEDTE(I,1);
  BES=BESTEMMING(I,1);
  DO WHILE ( WEER > 0 & VAK(I,1) > 0 );
/* RECHTSAFSLAAND VERKEER MAG VERTREKKEN. */
  VERTREKCYC(I)=VERTREKCYC(I)+1;
  VERTREKSOM(I)=VERTREKSOM(I)+1;
  VERTREK(BES)=VERTREK(BES)+1;
  VAK(I,1)=VAK(I,1)-1;
  WEER=WEER-1;
END;END;

```

```

ENDB2: END;
/* IN DEZE DO-LOOP WORDT OP DE KRUISPUNTEN */
/* HET RECHTDOORGAAND VERKEER BEHANDELD. */
B3: DO I=1 TO AANTAL;
  IF KLASSE(I)<2 THEN GOTO ENDB3;
  IF LICHT(I,2)=GROEN & VAK(I,2) > 0

```


PROCEDURE OPTIONS (MAIN);

```
    THEN DO; BES=BESTEMMING(I,2);
      IF VORM(I)≠4
        THEN DO; WEER=BREEDTE(I,2);
          DO WHILE (WEER > 0 & VERTREK(BES) <
            BREEDTE(I,2) & VAK(I,2) > 0);
            VAK(I,2)=VAK(I,2)-1;
            VERTREKCYC(I)=VERTREKCYC(I)+1;
            VERTREKSOM(I)=VERTREKSOM(I)+1;
            VERTREK(BES)=VERTREK(BES)+1;
            WEER=WEER-1;
          END;
        END;
      ELSE DO; IF VERTREK(BES) < 2 & VAK21(I) > 0
        THEN DO; VAK21(I) =VAK21(I)-1;
          VAK(I,2)=VAK(I,2)-1;
          VERTREKCYC(I)=VERTREKCYC(I)+1;
          VERTREKSOM(I)=VERTREKSOM(I)+1;
          VERTREK(BES)=VERTREK(BES)+1;
        END;
        IF VERTREK(BES)<2 & VAK23(I) > 0
          THEN DO; VAK23(I)=VAK23(I)-1;
            VAK(I,2)=VAK(I,2)-1;
            VERTREKCYC(I)=VERTREKCYC(I)+1;
            VERTREKSOM(I)=VERTREKSOM(I)+1;
            VERTREK(BES)=VERTREK(BES)+1;
          END;
        END;      END;      END;
ENDB3: END;
/* IN DEZE DO-LOOP WORDT ALLE VERTREK DEFINITIEF UITGEVOERD. */
B4: DO I=1 TO AANTAL;
  IF VERTREK(I)>0
    THEN DO; WEER=VERTREK(I); VERTREK(I)=0;
      RIJDEND(I)=RIJDEND(I)+WEER;
      IF KLASSE(I)=VERB
        THEN DO;
B4A: NORM=-6;
      DO J=1 TO 12;
        CALL RANDOM(INT,REAL); NORM=NORM+REAL;
      END;
      SNELH=ABS(SNELHEID(I)+NORM*SPREIDING(I));
      RIJTIJD=3.6*LENGTE(I)/(SNELH*TAU);
      PLAATS=RIJTIJD;
      DO WHILE (PLAATS>POS); PLAATS=PLAATS-CTAU;
    END;
      PLAATS=PLAATS+TACT;
B4B: IF PLAATS > POS THEN PLAATS=PLAATS-POS;
      IF B(I,PLAATS)=0
        THEN B(I,PLAATS)=1;
```

PROCEDURE OPTIONS (MAIN);

```

                                ELSE DO; IF B(I,PLAATS) < RIJSTROKEN(I)
                                    THEN B(I,PLAATS)=B(I,PLAATS)+1;
                                    ELSE DO; PLAATS=PLAATS+1;
                                                VERTRAGING(I)=
                                                VERTRAGING(I)+1;
                                                GOTO B4B;
                                END;      END;
                                WEER=WEER-1; IF WEER > 0 THEN GOTO B4A;
                                END;      END;
ENDB4: END;
/* EINDE VAN DE VERKEERSVERWERKING TIJDENS DIT ACTIETIJDSTIP. */
/* WACHTTIJD TELLINGEN WORDEN UITGEVOERD. */
    IWTOT=0;
C1:   DO I=1 TO AANTAL;
        IF KLASSE(I)<2 THEN GOTO ENDC1;
        IF AANKOMST1(I) =0 THEN JW=0; ELSE JW=1;
        IF AANKOMST2(I) =0 THEN JW=JW+1;
        JW=JW+VAK(I,1)+VAK(I,2)+VAK(I,3)+FILE1(I)+FILE2(I)
                                                +BUFFER(I);

        WACHTEND(I)=JW;
        WACHTTIJD(I)=WACHTTIJD(I)+JW;
        IWTOT=IWTOT+JW;
ENDC1: END;
    IWTOTAAL=IWTOTAAL+IWTOT;
/* INDIEN NODIG WORDEN TUSSEN RESULTATEN GEPRINT. */
    IF (CYCLUS=PRC)&(TFASE=PRT) THEN DO;
                                                G=1; NP=PRG;
                                                END;

    IF (G=0)|(NG<1) THEN GOTO SIM;
    IF (NP+PRS) < 1 THEN NP=PRG;
    NP=NP-1;
    IF NP < 0 THEN GOTO SIM;
    NG=NG-1;
    IF REGEL >52 THEN DO; CALL KOPP(BLZ);  REGEL=5;
                        END;
    PUT SKIP(2) EDIT('VERKEERSGEGEVENS NA CYCLUS__ACTIETIJD:',
CYCLUS,' __',TFASE)(X(11),A(40),F(4),A(3),F(3));  PUT SKIP;
    REGEL=REGEL+3; NK=4;
/* IN DEZE DO-LOOP WORDEN DE AANKOMSTTABELLEN VAN DE VERBINDINGS */
/* WEGEN GEPRINT MITS DEZE NIET LEEG ZIJN. */
D1:   DO I=1 TO AANTAL;
        IF KLASSE(I)=VERB & RIJDEND(I)>0
        THEN DO; IF NK=4
                    THEN DO;IF REGEL+4>60 THEN DO; CALL KOPP(BLZ);
                                REGEL=5;
                                END;      END;
                    PUT SKIP EDIT(I,'')(F(3),A(2));

```

PROCEDURE OPTIONS (MAIN);

```

DO J=3,2,1;
  IF LICHT(I,J)=ROOD THEN PUT EDIT('R')(A);
  ELSE PUT EDIT('G')(A);
END; PUT EDIT('  ')(A);
IF TACT=POS
THEN PUT EDIT((B(I,J) DO J=POS, 1 TO POS-1))(F(1));
ELSE PUT EDIT((B(I,J) DO J=TACT1 TO POS,1 TO TACT))
(F(1)); PUT EDIT(RIJDEND(I))(COLUMN(112),F(4));
REGEL=REGEL+1; NK=NK-1;
IF NK=0
THEN DO; NK=4;
  IF REGEL < 60 THEN DO; PUT SKIP;
  REGEL=REGEL+1;
END;
END; END; END;

```

ENDD1: END;

```

IF REGEL < 60 THEN DO; PUT SKIP;
  REGEL=REGEL+1;
END;

```

NK=4;

/* IN DEZE DO-LOOP WORDT DE GEHELE NETWERK-STATUS UITGEPRINT. */

```

D2: DO I=1 TO AANTAL;
  IF KLASSE(I) > 1
  THEN DO; IF NK=4
  THEN DO; IF REGEL+4 > 60 THEN DO; CALL KOPP(PLZ);
  REGEL=5;
  END; END;
  PUT SKIP EDIT(I,'  ')(F(3),A(2));
  DO J=3,2,1;
  IF LICHT(I,J)=ROOD THEN PUT EDIT('R')(A);
  ELSE PUT EDIT('G')(A);
END;
  PUT EDIT((B(I,J) DO J=TACT1,TACT2,TACT3),
  WACTEND(I),BUFFER(I),(VAK(I,J) DO J=3,2,1
  ))(X(3),3(F(1)),F(6),X(1),4(F(5)));
  IF VORM(I)=4 THEN PUT EDIT(VAK23(I),VAK21(I),'  ')
  (2(F(5)),A(3));
  ELSE PUT EDIT('  ')(A(13));
  IF (VORM(I) < 5) & (VORM(I) ≠ 2)
  THEN DO; CALL PR_AK(AANKOMST2(I));
  PUT EDIT(FILE2(I),AFSLAAND2(I),'  ')
  (2(F(6)),A(4));
  END;
  ELSE PUT EDIT('  ')(A(17));
  IF VORM(I) < 5
  THEN DO; CALL PR_AK(AANKOMST1(I));
  PUT EDIT(FILE1(I),AFSLAAND1(I))(2(F(6)));
  END;

```

PROCEDURE OPTIONS (MAIN);

```

        ELSE PUT EDIT ('                ')(A(13));
        PUT EDIT (WACHTTIJD(I))(F(8));
        IF KLASSE(I)=VERB
        THEN PUT EDIT(VERTRAGING(I),VERTREKSOM(I),STOPS(I),
                RIJDEND(I))(F(5),3(F(6)));
        ELSE PUT EDIT(VERTREKSOM(I),STOPS(I))(X(5),2(F(6)));
        REGEL=REGEL+1; NK=NK-1;
        IF NK=0 THEN DO; NK=4;
                IF REGEL < 60 THEN DO; PUT SKIP;
                        REGEL=REGEL+1;
        END;
        END;
        END;
        END;
ENDD2: END;
        IF REGEL < 60 THEN DO; PUT SKIP;
                REGEL=REGEL+1;
        END;
        GOTO SIM;
PRINT_ALG:
/*  NA AFLOOP VAN DE SIMULATIE WORDEN ALLE EINDRESULTATEN          */
/*  VERZAMELD EN AFGEDRUKT.                                         */
        BLZ=BLZ+1;
        PUT PAGE EDIT (' OVERZICHT VAN DE SIMULATIE RESULTATEN NA',CSIM,' PERI
        ODEN      =' ,CSIM*CTAU,' ACTIETIJDSTIPPEN.')(A,F(5),A,F(8),A);
        PUT EDIT ('BLAD',BLZ)(X(22),A,F(4));
        J=CSIM*CSEC; I=FLOOR(J/60); J=J-60*I;
        PUT SKIP (2) EDIT (' DIT KOMT OVEREEN MET',I,' MIN.',J,' SEC.')(
                (A,F(7),A,F(5),A);PUT SKIP (2) EDIT(' VOORAF ZIJN ',
                PRE,' INLOOP-PERIODEN UITGEVOERD.')(A,F(2));
        IR=0; IS=0; IT=0; IU=0; IV=0; IW=0; IX=0; IY=0; IZ=SUM(VERTRAGING);
        DO I=1 TO AANTAL;
                IF KLASSE(I)=ING THEN DO; IR=IR+RIJDEND(I); IT=IT+WACHTEND(I);
                        IV=IV+STOPS(I); IX=IX+WACHTTIJD(I);
                END;
                ELSE IF KLASSE(I)=VERB
                        THEN DO; IS=IS+RIJDEND(I); IU=IU+WACHTEND(I);
                                IW=IW+STOPS(I); IY=IY+WACHTTIJD(I);
        END;
        END;
        PUT SKIP (3) EDIT ('+-----+-----+-----+')(X(61),A);
        PUT EDIT ('| INGAANDE |VERBINDINGS| TOTAAL |',
                '| WEGEN | WEGEN | |',
                '+-----+-----+-----+')(SKIP,X(61),A);
        PUT SKIP EDIT (' AANTAL VOERTUIG-'STOPS' GESOMMEERD OP GROENBEGI
        NTIJDSTIPPEN:',IV,' |',IW,' |',IV+IW,' |')(A,3(F(8),A));
        PUT SKIP EDIT (' WACHTTIJD STILSTAAND BEREKEND IN VOERTUIG-ACTIETI
        JDSTIPPEN :',IX,' |',IY,' |',IX+IY,' |')(A,3(F(8),A));
        PUT SKIP EDIT (' VERTRAGING RIJDEND OPGELOPEN DOOR DE PLAATINGS-P
        ROCEDURE : |',IZ,' |',IZ,' |')(A,2(F(8),A));
        PUT SKIP EDIT (' SOM VAN WACHTTIJD STILSTAAND EN VERTRAGING RIJDEN
```

PROCEDURE OPTIONS (MAIN);

```

D      :',IX,'  |',IY+IZ,'  |',IX+IY+IZ,'  |')(A,3(F(8),A));
PUT SKIP EDIT (': | | | |')(X(61),A);
PUT SKIP EDIT(' AAN HET EINDE VAN DE SIMULATIE NOG RIJDENDE VOERTU
IGEN   :',IR,'  |',IS,'  |',IR+IS,'  |')(A,3(F(8),A));
PUT SKIP EDIT(' AAN HET EINDE VAN DE SIMULATIE STILSTAANDE VOERTU
IGEN   :',IT,'  |',IU,'  |',IT+IU,'  |')(A,3(F(8),A));
PUT SKIP EDIT ('+-----+-----+-----+')(X(61),A);
REGEL=19;
      CALL KOPV(BLZ,REGEL);
IS=0; IU=0; IV=0; IW=0; IX=0; IY=0; IZ=0;
E1: DO I=1 TO AANTAL;
      IF KLASSE(I)=VERB THEN
      DO; IR=MAX(GROENFILE(I,1),GROENFILE(I,2));
          IQ=MAX(PERIODEVERTREK(I,1),PERIODEVERTREK(I,2));
          IS=IS+VERTREKSOM(I);
          IT=VERTREKSOM(I)+WACHTEND(I)+RIJDEND(I); IU=IU+IT;
          IV=IV+WACHTEND(I); IW=IW+STOPS(I); IX=IX+WACHTTIJD(I);
          IY=IY+VERTRAGING(I); IZ=WACHTTIJD(I)+VERTRAGING(I);
          DO J=3 TO CSIM; IR=MAX(IR,GROENFILE(I,J));
              IQ=MAX(IQ,PERIODEVERTREK(I,J));
          END;
          IF IQ=0 THEN A=100;
              ELSE A=IR/IQ;
          PUT SKIP EDIT ('|',I,'  |',IT,'  |',VERTREKSOM(I),
              ' |',IR,'  |',A,'  |',WACHTEND(I),'  |',STOPS(I)
              ),' |',WACHTTIJD(I),'  |',VERTRAGING(I),'  |',IZ,'  |'
              (A,F(7),A,3(F(8),A),X(4),F(4,1),A,5(F(8),A));
          REGEL=REGEL+1; IF REGEL=58 THEN CALL KOPV(REGEL,BLZ);
ENDEL: END;END;
PUT SKIP EDIT ('| ',(9)' |', ' |')(A);
PUT SKIP EDIT ('| TOTAAL |',IU,'  |',IS,'  |', ' |
|',IV,'  |',IW,'  |',IX,'  |',IY,'  |',IX+IY,'  |'
(A,2(F(8),A),A,5(F(8),A));
PUT SKIP EDIT ('+-(9)-----+', '-----+')(3(A));
REGEL=REGEL+3;
      CALL KOPI(BLZ,REGEL);
IS=0; IU=0; IV=0; IW=0; IX=0;
E2: DO I=1 TO AANTAL;
      IF KLASSE(I)=ING THEN
      DO; IR=MAX(GROENFILE(I,1),GROENFILE(I,2));
          IQ=MAX(PERIODEVERTREK(I,1),PERIODEVERTREK(I,2));
          IS=IS+VERTREKSOM(I);
          IT=VERTREKSOM(I)+WACHTEND(I)+RIJDEND(I); IU=IU+IT;
          IV=IV+WACHTEND(I); IW=IW+STOPS(I); IX=IX+WACHTTIJD(I);
          DO J=3 TO CSIM; IR=MAX(IR,GROENFILE(I,J));
              IQ=MAX(IQ,PERIODEVERTREK(I,J));
          END;
      END;

```

PROCEDURE OPTIONS (MAIN);

```

        IF IQ=0 THEN A=100;
            ELSE A=IR/IQ;
        PUT SKIP EDIT ('|',I,'      |',IT,'      |',VERTREKSOM(I),
            '      |',IR,'      |',A,'      |',WACHTEND(I),'      |',STOPS(I)
            ),'      |',WACHTTIJD(I),'      |')
            (A,F(7),A,3(F(8),A),X(4),F(4,1),A,3(F(8),A));
        REGEL=REGEL+1; IF REGEL=58 THEN CALL KOPI(REGEL,BLZ);
ENDE2: END;END;
PUT SKIP EDIT ('|',(8)'      |')(A);
PUT SKIP EDIT ('|      TOTAAL      |',IU,'      |',IS,'      |',
            |',IV,'      |',IW,'      |',IX,'      |')(A,2(F(8),A),A,3(F(8),A));
PUT SKIP EDIT ('+-' , (8) '-----+')(A,A);
REGEL=REGEL+3;
        CALL KOPU(BLZ,REGEL);
IR=0; IS=0;
E3: DO I=1 TO AANTAL;
        IF KLASSE(I)=UITG THEN
            DO; IR=RIJDEND(I); IS=IS+IR;
                PUT SKIP EDIT ('|',I,'      |',IR,'      |')(A,F(7),A,F(8),A);
                REGEL=REGEL+1; IF REGEL=58 THEN CALL KOPU(REGEL,BLZ);
ENDE3: END;END;
PUT SKIP EDIT ('|      |      |')(A);
PUT SKIP EDIT ('|      TOTAAL      |',IS,'      |')(A,F(8),A);
PUT SKIP EDIT ('+-----+-----+')(A);
REGEL=REGEL+3;
PRINT_SPEC:
/* MET BEHULP VAN TOEGEVDEGDE DATA-KAARTEN KAN OPgegeVEN WORDEN */
/* VAN WELKE WEGVAKKEN EEN MEER UITGEBREID VERKEERS-OVERZICHT */
/* VERLANGD WORDT. PER TWEE WEGVAKKEN WORDT DAARNA DE AANWEZIGE */
/* INFORMATIE AFGEDRUKT. */
        GET LIST (WEGA); IF WEGA=0 THEN GOTO ENDPRO; GET LIST (WEGB);
        PUT PAGE; BLZ=BLZ+1;
            PUT EDIT('| RESULTATEN VAN      |',V = AANTAL VERTROKKEN VO
ERTUIGEN TIJDENS DE AFGELOPEN GROENFASE.', 'BLAD',BLZ)(X(4),A,X(6),A,
X(18),A,F(4)); PUT SKIP EDIT ('| WEG      | WEG      |',
            |'W = AANTAL WACHTENDE VOERTUIGEN WANN
EER HET RECHTDOORLICHT GROEN WORDT.')(X(4),A,X(6),A);
        PUT SKIP EDIT('      |',WEGA,'      |',WEGB,'      |')(2(A,F(5)),A);
        PUT SKIP EDIT('CYC-|      |      |')(A);
        REGEL=4; CALL KOPW(BLZ,REGEL);
F1: DO CYCLUS=1 TO CSIM;
        REGEL=REGEL+1; IF REGEL>58 THEN CALL KOPW(BLZ,REGEL);
        PUT SKIP EDIT(CYCLUS,PERIODEVERTREK(WEGA,CYCLUS),GROENFILE(WEGA,
            CYCLUS),PERIODEVERTREK(WEGB,CYCLUS),GROENFILE(
            WEGB,CYCLUS))(F(3),2(X(1)-2(F(4))));
            WK=GROENFILE(WEGA,CYCLUS); NK=1;
            WG=GROENFILE(WEGB,CYCLUS); NG=2;

```

PROCEDURE OPTIONS (MAIN);

```

IF WK>WG THEN DO; WG=WK; NG=1; NK=2;
                    WK=GRDENFILE(WEGB,CYCLUS);
                    END;
IF WK>95 THEN DO; PUT EDIT(' |')(A(3)); GOTO ENDF1; END;
IF WG>95
THEN DO; IF WK=0
          THEN PUT EDIT(NK)(X(2),F(1));
          ELSE DO; WK=WK-1;
                  PUT EDIT(' |',NK)(A(3),X(WK),F(1));
          END;
          GOTO ENDF1;
END;
IF WK=WG
THEN DO; IF WK<2 THEN DO; WK=WK+1;
          PUT EDIT(NK,'.',NG)
          (X(WK),F(1),A(1),F(1));
          END;
          ELSE DO; WK=WK-2;
                  PUT EDIT(' |',NK,'.',NG)
                  (X(2),A(1),X(WK),F(1),A(1),F(1));
          END;
          GOTO ENDF1;
END;
WV=WG-WK-1;
WK=WK-1;
IF WK=-1
THEN PUT EDIT(NK,NG)(X(2),F(1),X(WV),F(1));
ELSE PUT EDIT(' |',NK,NG)(A(3),X(WK),F(1),X(WV),F(1));
ENDF1: END;
PUT SKIP EDIT('-----+',(11)'-----')(A);
IF REGEL>43 THEN DO; BLZ=BLZ+1; REGEL=1;
                    PUT PAGE EDIT ('BLAD',BLZ)(X(112),A,F(4));
                    END;
PUT SKIP(3) EDIT('+-----+-----+')(X(23),A);
PUT SKIP EDIT(' | WEG',WEGA,' | WEG',WEGB,' |')
              (X(23),A,F(4),A,F(4),A);
PUT SKIP EDIT('+-----+-----+')(X(23),A);
PUT EDIT (' AANTAL 'STOPS' :',STOPS(WEGA),' |',
STOPS(WEGB),' |',' WACHTTIJD STILSTAAND :',WACHTTIJD(WEGA),' |',
',WACHTTIJD(WEGB),' |',' VERTRAGING RIJDEND :',VERTRAGING(WEGA),
', VERTRAGING(WEGB),' |',' WACHTTIJD TOTAAL :',
VERTRAGING(WEGA)+WACHTTIJD(WEGA),' |',
VERTRAGING(WEGB)+WACHTTIJD(WEGB),' |',' STILSTAAND :',
,WACHTEND(WEGA),' |',WACHTEND(WEGB)-', |',' RIJDEND :',
RIJDEND(WEGA),' |',RIJDEND(WEGB),' |',' VERTROKKEN :',
VERTREKSOM(WEGA),' |',VERTREKSOM(WEGB),' |')
(7(SKIP(2),A,2(F(9),A))); REGEL=REGEL+17;

```

PROCEDURE OPTIONS (MAIN);

```
IF REGEL+5+CTAU>60
THEN DO; REGEL=1; BLZ=BLZ+1;
      PUT PAGE EDIT ('BLAD',BLZ)(X(112),A,F(4));
      END;
PUT SKIP EDIT(' FASE- | | AANKOMST VERDELING WEG',WEGA,'.',
              ' FASE- | | AANKOMST VERDELING WEG',WEGB,'.')
      (A,F(4),A,X(18),A,F(4),A);
PUT SKIP EDIT(' TIJD | |', ' TIJD | |')(2(A(12),X(48)));
PUT SKIP EDIT('<+>-----+',(5)'-----', '<+>-----+',(5)'-----')
      (A,A,X(2));      REGEL=REGEL+3;
F2: DO TFASE=1 TO CTAU;
      PUT SKIP; REGEL=REGEL+1;
      IF REGEL>60
      THEN DO; BLZ=BLZ+1; REGEL=2;
              PUT EDIT('BLAD',BLZ)(X(112),A,F(4)); PUT SKIP;
              END;
      DO I=WEGA,WEGB;
          DO J=3,2,1;
              IF TFASE=GROENBEGIN(I,J) THEN LICHT(I,J)=GROEN;
              IF TFASE=ROODBEGIN(I,J) THEN LICHT(I,J)=ROOD;
              IF LICHT(I,J)=ROOD THEN PUT EDIT('R')(A);
              ELSE PUT EDIT('G')(A);
          END;
          K=AANK_VERDELING(I,TFASE); G=47-K;
          PUT EDIT(TFASE,K,' |')(F(3),F(4),A);
          IF K=0 THEN PUT EDIT ('')(X(47),A);
          ELSE DO; IF K>48
                  THEN PUT EDIT((47)*', '>')(A);
                  ELSE DO; DO J=1 TO K;
                          PUT EDIT('*')(A);
                  END;
                  IF K < 48
                  THEN PUT EDIT ('')(X(G),A);
      END;
      END;      END;      END;
ENDF2: END;
IF REGEL < 59
THEN DO; PUT SKIP EDIT ('<|>-----+',(5)'-----', '<|>-----+',
                        (5)'-----')(A,A,X(2));
      PUT SKIP EDIT(VERTREKSJM(WEGA)+WACHTEND(WEGA),
                    VERTREKSOM(WEGB)+WACHTEND(WEGB))(F(10),X(50));
      END;
      GOTO PRINT_SPEC;
RICHT: PROCEDURE(RECHTS,NIETLINKS,RI);
/* DE PROCEDURE RICHT BEPAALT DMV LOTING EEN
/* VERVOLGRICHTING VOOR EEN VOERTUIG
/* DAT EEN KROISPUNT NADERT.
DECLARE (RECHTS,NIETLINKS) BINARY FLOAT;
```


PROCEDURE OPTIONS (MAIN);

```
    DECLARE RI                BINARY FIXED(15);
    CALL RANDOM(INT,REAL);
    IF REAL < RECHTS THEN RI=1;
    ELSE DO; IF REAL > NIETLINKS THEN RI=3;
    ELSE RI=2;
    END;
    RETURN;
END RICHT;
VUL:  PROCEDURE(FILE,AFSLAAND,RI);
/*    DE PROCEDURE VUL BEPAALT DMV LOTING EEN                */
/*    VERVOLGRICHTING VOOR EEN VOERTUIG DAT IN              */
/*    EEN FILE STAAT, BESTAANDE UIT RECHTDOORGAAND          */
/*    EN NAAR EEN ZIJDE AFBUIGEND VERKEER.                  */
    DECLARE (FILE,AFSLAAND,RI)    BINARY FIXED(15);
    DECLARE FRACTIEAF              BINARY FLOAT;
    IF AFSLAAND=0
    THEN RI=2;
    ELSE DO; IF FILE=AFSLAAND
    THEN RI=13;
    ELSE DO; CALL RANDOM(INT,REAL);
    FRACTIEAF=AFSLAAND/FILE;
    IF REAL > FRACTIEAF THEN RI=2;
    ELSE RI=13;
    END;    END;
    RETURN;
END VUL;
PR_AK: PROCEDURE(AK);
/*    DE PROCEDURE PR_AK PRINT DE INHOUD VAN EEN AANKOMSTVAK. */
    DECLARE AK BINARY FIXED(15);
    IF AK=0
    THEN PUT EDIT (AK)(F(1));
    ELSE DO; IF AK=2
    THEN PUT EDIT ('|')(A);
    ELSE DO; IF AK=1 THEN PUT EDIT ('>')(A);
    ELSE PUT EDIT ('<')(A);
    END;    END;
    RETURN;
END PR_AK;
KOPI: PROCEDURE(BLZ,REGEL);
/*    DE PROCEDURE KOPI PRINT HET KOPJE BOVEN DE EINDRESULTATEN */
/*    VAN DE INGAANDE WEGEN.                                        */
    DECLARE (BLZ,REGEL) BINARY FIXED(15);
    IF REGEL>25 THEN DO; BLZ=BLZ+1; REGEL=1;
    PUT PAGE EDIT ('BLAD',BLZ)(X(112),A,F(4));
    END;
PUT SKIP (3);
PUT SKIP EDIT ('+-',(8)'-----+')(A,A);
```

PROCEDURE OPTIONS (MAIN);

```

PUT SKIP EDIT ('| ',(4)' |', ' MINIMAAL | FILE |',
(2)' |')(A);
PUT SKIP EDIT('| WEGVAK NR. | INGEKOMEN | UITGEGAAN | MAXIMUM | BENOD
IGDE | NA AFLOOP | ''STOPS'' | WACHTTIJD |')(A);
PUT SKIP EDIT('| INGAAND | | FILE | CY
CLI | SIMULATIE | | STILSTAAND |')(A);
PUT SKIP EDIT('+-',(8)'-----+')(A,A);
REGEL=REGEL+8;
RETURN;

END KOPI;
KOPP: PROCEDURE(BLZ);
/* DE PROCEDURE KOPP PRINT HET KOPJE BOVEN DE TUSSENRESULTATEN */
/* TIJDENS DE SIMULATIE. */
DECLARE BLZ BINARY FIXED(15);
BLZ=BLZ+1;
PUT PAGE EDIT ('BLAD',BLZ)(X(112),A(4),F(4));
PUT SKIP EDIT(' NR LICHT TABEL STIL BUF VAK VAK VAK VAK VA
K AK2 FILE AFSL AK1 FILE AFSL WACHT VERTR VERTR STOPS RIJ')(A);
PUT SKIP EDIT(' 321 123 STAAND 3 2 1 23 2
1 2 2 1 1 TIJD AGING OKKEN DEND')(A);
PUT SKIP EDIT('---+<|>+-----+-',(8)'----+',(2)'-----+',
(7)'----+-',(7)'----+')(A);

PUT SKIP;
RETURN;

END KOPP;
KOPS: PROCEDURE(BLZ,REGEL);
/* DE PROCEDURE KOPS PRINT HET KOPJE BOVEN DE INGEVOERDE GEGEVENS. */
DECLARE (BLZ,REGEL) BINARY FIXED(15);
IF REGEL>40
THEN DO; BLZ=BLZ+1; REGEL=1;
PUT PAGE EDIT ('BLAD',BLZ)(X(112),A,F(4));
END; PUT SKIP EDIT(
'SEC SEC','ACTIE ACTIE',' NR RIJ VORM BES3 BES2 BE
S1 TEGL %-L %R MAX-VAKKEN MF MBUF GROENBEGINTIJD ROODBEGINTIJD BREE
DTE INT LANG SNEL SPR')(X(69),A,SKIP,X(68),A,SKIP,A);
PUT SKIP EDIT('3','2','1','3','2','1','3','2','1','3 2 1 B')
(X(41),3(X(3),A(1)),X(8),6(X(4),A(1)),X(2),A(7));
PUT SKIP EDIT((3)'+---',(5)'-+---',(6)''+---',(7)''-+---',(2)'-----+',
-',(2)'-+---')(A); PUT SKIP; REGEL=REGEL+6;
RETURN;

END KOPS;
KOPU: PROCEDURE(BLZ,REGEL);
/* DE PROCEDURE KOPU PRINT HET KOPJE BOVEN DE EINDRESULTATEN */
/* VAN DE UITGAANDE WEGEN. */
DECLARE (BLZ,REGEL) BINARY FIXED(15);
IF REGEL>35 THEN DO; BLZ=BLZ+1; REGEL=1;
PUT PAGE EDIT ('BLAD',BLZ)(X(112),A,F(4));

```

PROCEDURE OPTIONS (MAIN);

END;

```
PUT SKIP (3) EDIT('+-+-----+-----+',
                '| | WEGVAK NR. | INGEKOMEN |',
                '| | UITGAAND | |',
                '+-----+-----')(SKIP,A);
```

REGEL=REGEL+8;

RETURN;

END KOPU;

KOPV: PROCEDURE(BLZ,REGEL);

/* DE PROCEDURE KOPV PRINT HET KOPJE BOVEN DE EINDRESULTATEN */
/* VAN DE VERBINDINGS WEGEN. */

DECLARE (BLZ,REGEL) BINARY FIXED(15);

IF REGEL>35 THEN DO; BLZ=BLZ+1; REGEL=1;

PUT PAGE EDIT ('BLAD',BLZ)(X(112),A,F(4));

END;

PUT SKIP EDIT('+-',(9)'-----+', '-----+')(SKIP(3),3(A));

```
PUT SKIP EDIT ('| |(4)' |,' MINIMAAL | FILE |',
(3)' |,' |')(A);
```

```
PUT SKIP EDIT ('| WEGVAK NR. | INGEKOMEN | UITGEGAAN | MAXIMUM | BENOD
IGDE | NA AFLOOP | 'STOPS' | WACHTTIJD |VERTRAGING |WACHTTIJD|')(A)
```

```
;PUT SKIP EDIT ('| VERBINDEND | | | FILE | CY
CLI | SIMULATIE | | STILSTAAND| RIJDEND | TOTAAL |')(A);
```

PUT SKIP EDIT('+-',(9)'-----+', '-----+')(A);

REGEL=REGEL+8;

RETURN;

END KOPV;

KOPW: PROCEDURE(BLZ,REGEL);

/* DE PROCEDURE KOPW PRINT HET KOPJE BOVEN DE TABEL VAN HET */
/* AANTAL WACHTENDE VOERTUIGEN TIJDENS DE GROENBEGINTIJDSTIPPEN. */

DECLARE (BLZ,REGEL) BINARY FIXED(15);

IF REGEL>40

THEN DO; BLZ=BLZ+1; REGEL=1;

PUT PAGE EDIT('CYC-| | |','BLAD',BLZ)
(A,X(88),A,F(4));

END;

PUT SKIP EDIT('LUS.| V1 W1 | V2 W2 |',(J DO J=10 BY 10 TO 90
))(A,9(X(8),F(2)));

PUT SKIP EDIT('-----+-----')(A); PUT EDIT((11)'-----+-----')(A);

REGEL=REGEL+2;

RETURN;

END KOPW;

ENDPRO: EX=EX-1; IF EX > 0 THEN GOTO RET;

END;

END NETSIM.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
				1	PRINT ON,NOGEN,NO&DATA
000000				2	RANDOM CSECT
000000				3	USING *,15
000000	9025 D01C		0001C	4	STM 2,5,28(13)
000004	9823 1000		00000	5	LM 2,3,0(1)
000008	5850 F030		00030	6	L 5,A
00000C	5C40 2000		00000	7	M 4,0(0,2)
000010	5D40 F034		00034	8	D 4,P
000014	5040 2000		00000	9	ST 4,0(0,2)
000018	8840 0007		00007	10	SRL 4,7
00001C	5A40 F02C		0002C	11	A 4,CHAR
000020	5040 3000		00000	12	ST 4,0(0,3)
000024	9825 D01C		0001C	13	LM 2,5,28(13)
000028	07FE			14	BR 14
00002A	0000				
00002C	40000000			15	CHAR DC F'1073741824'
000030	000041A7			16	A DC F'16807'
000034	7FFFFFFF			17	P DC F'2147483647'
				18	END

E OPTICNS (MAIN);

```

      IF KLASSE(I)=2 THEN PUT EDIT('      ING') (A);
      ELSE PUT EDIT('      VERB') (A);
      R=INTENS(I); V=R*CSEC*CSIM;
      PUT EDIT(V,(V*RICH(I,3)),(V*RICH(I,2)),(V*RICH(I,1)),(R*CSEC),(
R*CSEC*RICH(I,3)),(R*RICH(I,2)*CSEC),(R*RICH(I,1)*CSEC))
      (4(F(10)),X(8),4(F(6,1)));
ENDAAO: END;
PUT PAGE EDIT('          CAPACITEIT/OUR          CAPACITEIT/CYCLUS') (A);
PUT SKIP EDIT('KRUISING          IN          UIT          IN          UIT          ') (A);
PUT SKIP;
DO J=1 TO AKR; V=0; N=0;
  IC I=1 TO AANTAL;
  IF VAN(I)=J THEN V=V+INTENS(I);
  IF NAAR(I)=J THEN N=N+INTENS(I);
  END; PUT SKIP EDIT(J,V*3600,N*3600,V*CSEC,N*CSEC)
  (F(5),X(4),2(F(7)),X(6),2(F(7,1)));
END; V=0; N=0; J=0;
DO I=1 TO AANTAL;
  IF VAN(I)=0 THEN V=V+INTENS(I);
  IF NAAR(I)=0 THEN N=N+INTENS(I);
END;
PUT SKIP(3) EDIT('NETWERK          ',V*3600,N*3600,V*CSEC,N*CSEC)
  (A,2(F(7)),X(6),2(F(7,1)));
END INTEN;

```