

ONE

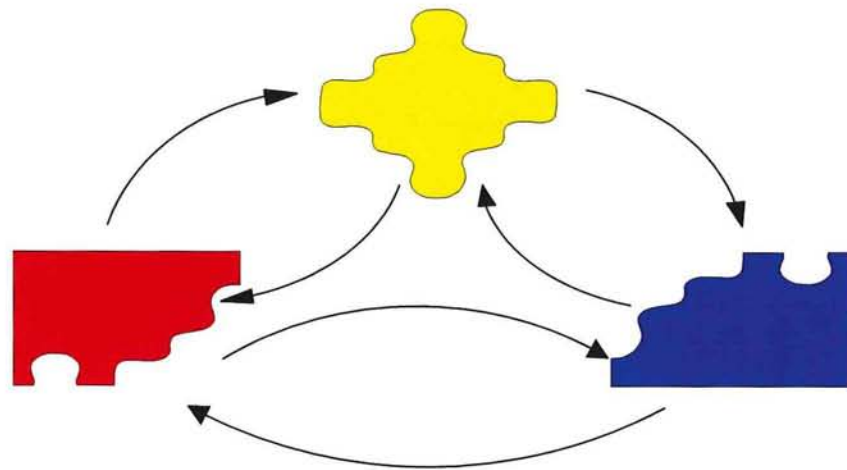
Afstudeeronderzoek  
Bijlagen eindrapport

# Een systeem dynamische verkenning van interregionale mobiliteit

7 februari 2002

J.J.C. Hennekam

---





# **Een systeem dynamische verkenning van interregionale mobiliteit**

**Ruimtelijke ontwikkeling van veertig  
Nederlandse Corop-gebieden**

**Bijlagen eindrapport  
Afstudeeronderzoek**

**J.J.C. Hennekam**

**Delft, 7 februari 2002**

Afstudeercommissie

prof. ir. F.M. Sanders  
Technische Universiteit Delft  
sectie Infrastructuurplanning

ir. P. van Eck  
Technische Universiteit Delft  
sectie Infrastructuurplanning

dr. ir. J. Cser  
Technische Universiteit Delft  
sectie Civieltechnische Informatica

dr. F.W.C.J. van de Vooren  
Universiteit Antwerpen  
vakgroep Transport en Ruimtelijke Economie  
Ministerie van Verkeer en Waterstaat  
directie Limburg

Sectie Infrastructuurplanning  
Subfaculteit Civiele Techniek  
Faculteit Civiele Techniek en Geowetenschappen  
Technische Universiteit Delft



## Voorwoord

Deze bijlagen behoren bij het eindrapport ‘*Systeem dynamische verkenning van interregionale mobiliteit*’, dat in het afstudeerkader verbonden aan de sectie Infrastructuurplanning van de faculteit der Civiele Techniek en Geowetenschappen is geschreven.

In deze bijlagen worden theoretische achtergronden, methodische aspecten, kwantitatieve analyseresultaten en overige praktische resultaten –waar in het eindrapport naar wordt verwezen– nader toegelicht. Ook de wiskundige formulering en kwantificering van het ontwikkelde model komen in deze bijlagen aan bod. Ter overzichtelijkheid zijn de bijlagen ingedeeld volgens de hoofdstukindeling van het eindrapport.

Delft, 7 februari 2002

Hans Hennekam



# INHOUDSOPGAVE

Voorwoord	i
<b>Bijlagen Hoofdstuk 2 Nederlandse schaalniveau's .....</b>	<b>1</b>
Bijlage 2.1 Samenhang Nederlandse NUTS-niveaus.....	3
Bijlage 2.2 Corop-regio's.....	4
Bijlage 2.3 Achtergrond regionale indelingen .....	7
Bijlage 2.3.1 Corop-gebieden .....	7
Bijlage 2.3.2 Economisch-geografische gebieden .....	8
Bijlage 2.3.3 Wgr-gebieden .....	8
Bijlage 2.3.4 Grootstedelijke agglomeraties .....	8
Bijlage 2.3.5 Het stadsgewest .....	9
Bijlage 2.3.6 Gemeentegroepen naar stedelijkheid .....	9
<b>Bijlagen Hoofdstuk 3 Methoden en technieken voor sociaal-economische analyse</b>	<b>11</b>
Bijlage 3.1 Methoden en technieken.....	13
Bijlage 3.2 Zwaartekrachtmodellen .....	13
Bijlage 3.3 Push-pull modellen .....	17
Bijlage 3.4 Input-Output modellen .....	18
Bijlage 3.5 Shift-Share .....	19
Bijlage 3.6 Lokatie-Quotiënt.....	20
Bijlage 3.7 Migratiestroom-modellen .....	21
Bijlage 3.8 Migratiesaldo-modellen.....	21
Bijlage 3.9 Cohort-Survival .....	22
<b>Bijlagen Hoofdstuk 4 Handleiding Array's in Stella .....</b>	<b>23</b>
Bijlage 4.1 Traditionele toepassingen van array's in Stella .....	25
Bijlage 4.1.1 Begripsdefiniëring array .....	25
Bijlage 4.1.2 Het idee: een regionaal bevolkingsmodel .....	25
Bijlage 4.1.3 Het voorbeeld: een multiregionaal bevolkingsmodel .....	26
Bijlage 4.1.4 De enkelvoudige array.....	28
Bijlage 4.1.5 De dubbele array.....	30
Bijlage 4.2 Creatieve toepassingen van array's in Stella .....	32
Bijlage 4.2.1 Hansen's Potential Model in Stella.....	32
Bijlage 4.2.2 Een double-constrained model in Stella .....	39
Bijlage 4.2.3 Overige modeltechnieken in Stella .....	40
<b>Bijlagen Hoofdstuk 5 Invoergegevens van Hansen's Potential Model.....</b>	<b>43</b>
<b>Bijlagen Hoofdstuk 6 Listing ontwikkeld Stella-model.....</b>	<b>49</b>
<b>Bijlagen Hoofdstuk 7 Module bevolking.....</b>	<b>123</b>
Bijlage 7.1 Multipliertafels module bevolking .....	125
Bijlage 7.1.1 Multipliertafels residentiële migratie .....	125
Bijlage 7.1.2 Multipliertafels arbeidsmigratie en pendel .....	126
Bijlage 7.1.3 Multipliertafels buitenlandse migratie bevolking.....	127
Bijlage 7.1.4 Multipliertafels grenspendel .....	129

Bijlage 7.2	Capaciteitsbeperking migratie .....	131
Bijlage 7.3	Desaggregatie van migratie .....	132
Bijlage 7.4	Modellering reistijdontwikkeling .....	133
<b>Bijlagen Hoofdstuk 8 Module Woningmarkt .....</b>		<b>135</b>
Bijlage 8.1	Multipliertafel module Woningmarkt .....	137
<b>Bijlagen Hoofdstuk 9 Module Bedrijven .....</b>		<b>139</b>
Bijlage 9.1	Multipliertafels module Bedrijven .....	141
Bijlage 9.1.1	Multipliertafels bedrijfsoprichting .....	141
Bijlage 9.1.2	Multipliertafel bedrijfsopheffing .....	142
Bijlage 9.1.3	Multipliertafels binnenlandse bedrijfsmigratie .....	142
Bijlage 9.1.4	Multipliertafels buitenlandse bedrijfsmigratie .....	144
Bijlage 9.2	Levenscyclus van bedrijven .....	146
Bijlage 9.3	Segmentatie bedrijven .....	147
<b>Bijlagen Hoofdstuk 10 Module Bedrijfsruimtemarkt .....</b>		<b>149</b>
<b>Bijlagen Hoofdstuk 12 Module Grondgebruik .....</b>		<b>153</b>
Bijlage 12.1	Multipliertafels module Grondgebruik .....	155
Bijlage 12.1.1	Multipliertafels aansturing woningvoorraad .....	155
Bijlage 12.1.2	Multipliertafels aansturing bedrijfsruimtevoorraad .....	156
<b>Bijlagen Hoofdstuk 13 Hindcast's .....</b>		<b>157</b>
Bijlage 13.1	Hindcast Nederland .....	159
Bijlage 13.1.1	Bevolking hindcast Nederland .....	159
Bijlage 13.1.2	Woningen hindcast Nederland .....	162
Bijlage 13.1.3	Arbeidsmarkt hindcast Nederland .....	163
Bijlage 13.1.4	Bedrijven hindcast Nederland .....	164
Bijlage 13.2	Modelfouten hindcast bevolking Utrecht .....	166
Bijlage 13.3	Modelfouten hindcast woningmarkt Utrecht .....	170
Bijlage 13.4	Modelfouten hindcast arbeidsmarkt Utrecht .....	172



## LIJST VAN FIGUREN

Figuur-Bijlage 2.1	Samenhang tussen Nederlandse NUTS-niveau's	3
Figuur-Bijlage 2.2	Nederlandse Corop-gebieden	4
Figuur-Bijlage 4.1	Causaal relatiediagram Nationale bevolkingsontwikkeling	25
Figuur-Bijlage 4.2	Modellering Nederlandse bevolkingsontwikkeling	26
Figuur-Bijlage 4.3	Causaal relatiediagram regionale bevolkingsontwikkeling	27
Figuur-Bijlage 4.4	Modellering regionale bevolkingsontwikkeling (Utrecht)	27
Figuur-Bijlage 4.5	Veertig losse modelstructuren	28
Figuur-Bijlage 4.6	Veertig modellen in enkelvoudige array	29
Figuur-Bijlage 4.7	Veertig gedesaggregeerde modellen in dubbele array	31
Figuur-Bijlage 4.8	Overzicht Stella-model van Hansen's Potential Model	35
Figuur-Bijlage 4.9	Sommeren in de array-structuur	36
Figuur-Bijlage 4.10	Invoergegevens voor een voorbeeld van een double-constrained zwaartekrachtmodel	39
Figuur-Bijlage 4.11	Iteratieve oplossingsprocedure van een double-constrained zwaartekrachtmodel	39
Figuur-Bijlage 4.12	Gebruikte matrices Hansen-model	47
Figuur-Bijlage 7.1	Multipliertafel AWONM_RESUITM	125
Figuur-Bijlage 7.2	Multipliertafel AWON_RESINM	125
Figuur-Bijlage 7.3	Correctiefactor residentiële migratie	126
Figuur-Bijlage 7.4	Multipliertafel ABM_UITARB	126
Figuur-Bijlage 7.5	Multipliertafel ABM_INARB	127
Figuur-Bijlage 7.6	Correctiefactor arbeidsmigratie en pendel	127
Figuur-Bijlage 7.7	Multipliertafel ABM_IMM_BEV	128
Figuur-Bijlage 7.8	Multipliertafel AWONM_IMM_BEV	128
Figuur-Bijlage 7.9	Multipliertafel ABM_EM_BEV	128
Figuur-Bijlage 7.10	Multipliertafel AWONM_EM_BEV	129
Figuur-Bijlage 7.11	Multipliertafel ABM_GRENSUITP	129
Figuur-Bijlage 7.12	Multipliertafel ABM_GRENSINP	130
Figuur-Bijlage 7.13	Idee endogene reistijdontwikkeling	134
Figuur-Bijlage 8.1	Multipliertafel WONHHBM	137
Figuur-Bijlage 9.1	Multipliertafel AOPRBESTBVM	141
Figuur-Bijlage 9.2	Multipliertafel ASTM	141
Figuur-Bijlage 9.3	Multipliertafel AOPHM	142
Figuur-Bijlage 9.4	Multiplier ABBM_UTM_BV	142
Figuur-Bijlage 9.5	Multipliertafel ABRUIMM_UTM_BV	143
Figuur-Bijlage 9.6	Multipliertafel ABBM_INM_BV	143
Figuur-Bijlage 9.7	Multipliertafel ABRUIMM_INM_BV	143
Figuur-Bijlage 9.8	Correctiefactor bedrijfsmigratie	144
Figuur-Bijlage 9.9	Multipliertafel ABBM_IMM_BV	144
Figuur-Bijlage 9.10	Multipliertafel ABRUIMM_IMM_BV	145
Figuur-Bijlage 9.11	Multipliertafel ABBM_EM_BV	145
Figuur-Bijlage 9.12	Multipliertafel ABRUIMM_EM_BV	145
Figuur-Bijlage 9.13	Levenscyclus van bedrijven	146
Figuur-Bijlage 9.14	Mogelijk onderscheid naar bedrijfssectoren	148
Figuur-Bijlage 10.1	Multipliertafel BRUIMBVBM	151
Figuur-Bijlage 10.2	Multipliertafel BRUIMBVSM	151
Figuur-Bijlage 12.1	Multipliertafel WONLANDBM	155
Figuur-Bijlage 12.2	Multipliertafel WONLANDSM	155
Figuur-Bijlage 12.3	Multipliertafel BRUIMLANDBM	156
Figuur-Bijlage 12.4	Multipliertafel BRUIMLANDSM	156

Figuur-Bijlage 13.1	Hindcast bevolking Nederland	159
Figuur-Bijlage 13.2	Hindcast natuurlijke aanwas Nederland	160
Figuur-Bijlage 13.3	Hindcast buitenlandse migratie Nederland	160
Figuur-Bijlage 13.4	Hindcast binnenlandse migratie Nederland	161
Figuur-Bijlage 13.5	Hindcast binnenlandse migratie naar reden Nederland	161
Figuur-Bijlage 13.6	Hindcast woningen Nederland	162
Figuur-Bijlage 13.7	Hindcast ontwikkelingen Nederlandse woningvoorraad	162
Figuur-Bijlage 13.8	Hindcast potentiële beroepsbevolking Nederland	163
Figuur-Bijlage 13.9	Hindcast arbeidsplaatsen Nederland	163
Figuur-Bijlage 13.10	Hindcast interregionale pendel Nederland	164
Figuur-Bijlage 13.11	Hindcast bedrijfsvestigingen Nederland	164

## LIJST VAN TABELLEN

Tabel-Bijlage 3.1	Sociaal-economische methoden en technieken naar tijd en ruimte	13
Tabel-Bijlage 4.1	Voorbeeldgegevens Hansen-model	34
Tabel-Bijlage 4.2	Afstands-/reistijdenmatrix	34
Tabel-Bijlage 4.3	Berekening $A_{ij}$ en $A_i$	34
Tabel-Bijlage 4.4	Berekening $D_i$	34
Tabel-Bijlage 4.5	Bepaling bevolkingsverdeling	35
Tabel-Bijlage 5.1	Invoer toestandsvariabelen van de Hansen-analyses	45
Tabel-Bijlage 5.2	Ontwikkeling invoer toestandsvariabelen van de Hansen-analyses	46
Tabel-Bijlage 13.1	Modelfouten hindcast bevolking Utrecht	166
Tabel-Bijlage 13.2	Modelfouten hindcast natuurlijke aanwas Utrecht	167
Tabel-Bijlage 13.3	Modelfouten hindcast binnenlandse migratie Utrecht	168
Tabel-Bijlage 13.4	Modelfouten hindcast buitenlandse migratie Utrecht	169
Tabel-Bijlage 13.5	Modelfouten hindcast woningvoorraad Utrecht	170
Tabel-Bijlage 13.6	Modelfouten hindcast ontwikkeling woningvoorraad Utrecht	171
Tabel-Bijlage 13.7	Modelfouten hindcast bedrijfsvestigingen Utrecht	172
Tabel-Bijlage 13.8	Modelfouten hindcast potentiële beroepsbevolking Utrecht	172

## **BIJLAGEN HOOFDSTUK 2**

### **NEDERLANDSE SCHAALNIVEAUS**



## Bijlage 2.1 Samenhang Nederlandse NUTS-niveaus

Het Centraal Bureau voor de Statistiek (CBS) onderscheidt voor Nederland verschillende *NUTS-niveaus* (regionale indelingen). De groepering van *Landsdelen* (*NUTS I*), *provincies* (*NUTS II*), en *Corop-gebieden* (*NUTS III*) is in figuur-Bijlage 2.1 weergegeven.

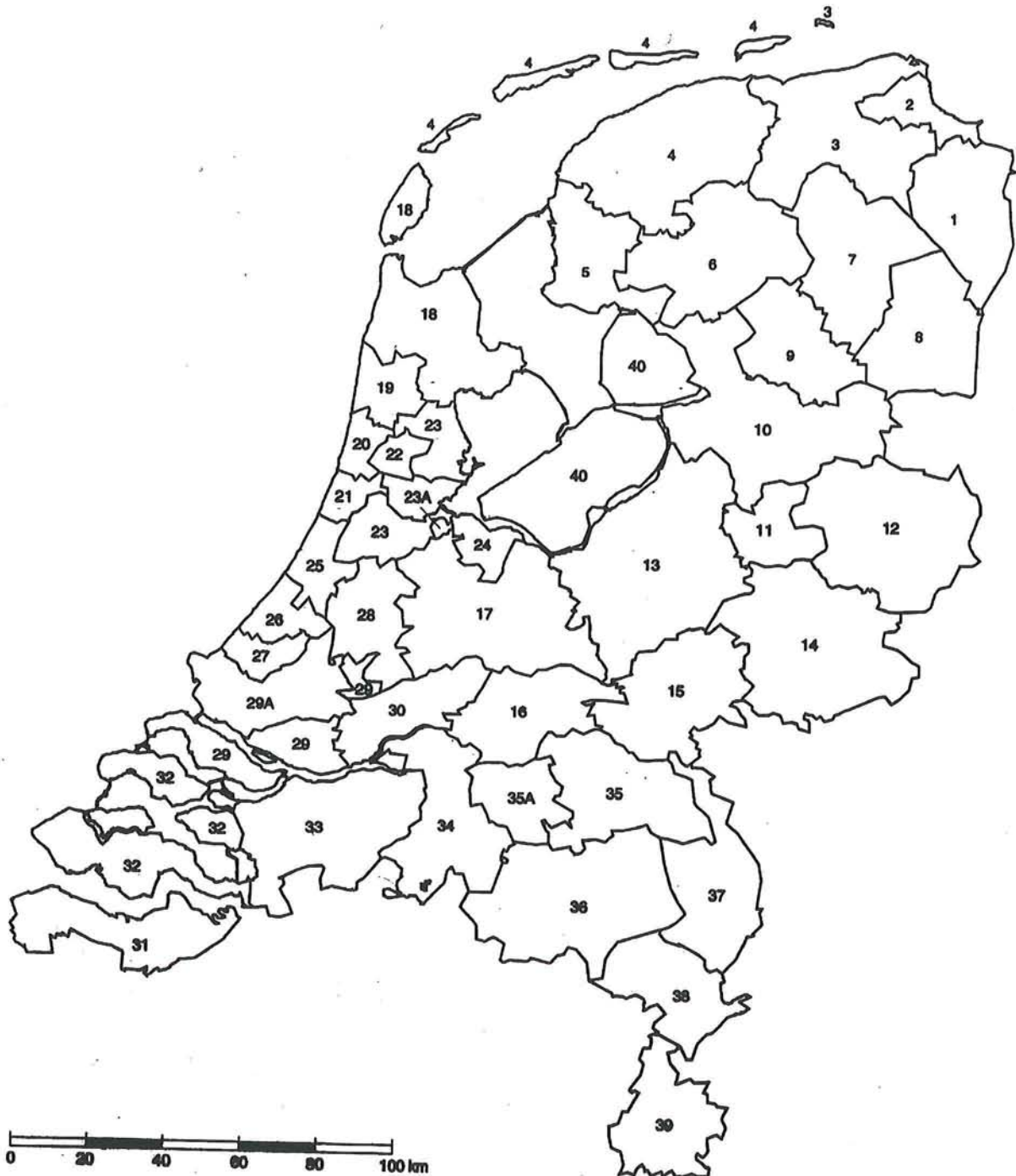
		landsdelen	provincies	Corop-regio's
Nederland	Noord-Nederland	Groningen	1	Oost-Groningen
			2	Delfzijl en omgeving
			3	Overig Groningen
		Friesland	4	Noord-Friesland
			5	Zuidwest-Friesland
			6	Zuidoost-Friesland
		Drenthe	7	Noord-Drenthe
			8	Zuidoost-Drenthe
			9	Zuidwest-Drenthe
	Oost-Nederland	Overijssel	10	Noord-Overijssel
			11	Zuidwest-Overijssel
			12	Twente
		Gelderland	13	Veluwe
			14	Achterhoek
			15	Arnhem/Nijmegen
		16	Zuidwest-Gelderland	
	Flevoland	40	Flevoland	
	West-Nederland	Utrecht	17	Utrecht
		Noord-Holland	18	Kop van Noord-Holland
			19	Alkmaar en omgeving
			20	IJmond
			21	Agglomeratie Haarlem
			22	Zaanstreek
			23*	Groot-Amsterdam
		24	Het Gooi en Vechtstreek	
		Zuid-Holland	25	Agglomeratie Leiden en Bollenstreek
			26	Agglomeratie 's-Gravenhage
			27	Delft en Westland
			28	Oost-Zuid-Holland
			29*	Groot-Rijnmond
			30	Zuidoost-Zuid-Holland
		Zeeland	31	Zeeuwsch-Vlaanderen
	32		Overig Zeeland	
	Zuid-Nederland	Noord-Brabant	33	West-Noord-Brabant
			34	Midden-Noord-Brabant
			35*	Noordoost-Noord-Brabant
			36	Zuidoost-Noord-Brabant
		Limburg	37	Noord-Limburg
			38	Midden-Limburg
39			Zuid-Limburg	

\*: Onderverdeeld in sub-gebieden

Figuur-Bijlage 2.1 Samenhang tussen Nederlandse NUTS-niveau's

## Bijlage 2.2 Corop-regio's

De indeling van Nederland in 40 Corop-regio's, met aparte vermelding van 3 subregio's, is weergegeven in figuur-Bijlage 2.2. Daaronder zijn per Corop de gemeenten op 1 januari 2000 gegeven.



Figuur-Bijlage 2.2 Nederlandse Corop-gebieden

(bron: Centraal Bureau voor de Statistiek)

<b>1. Oost-Groningen</b>	<b>7. Noord-Drenthe</b>	<b>13. Veluwe</b>	<b>16. Zuidwest-Gelderland</b>	Wester-Koggenland Wieringen Wieringermeer Wognum Zijpe
Bellingwedde Menterwolde Pekela Reiderland Scheemda Stadskanaal Veendam Vlagtwedde Winschoten	Aa en Hunze Assen Midden-Drenthe Noordenveld Tynaarlo	Apeldoorn Barneveld Ede Elburg Epe Ermelo Harderwijk Hatter Heerde Nijkerk Nunspeet Oldebroek Putten Scherpenzeel Voorst Wageningen	Buren Culemborg Dodewaard Echteld Geldermalsen Kesteren Lingewaal Maasdriel Neerijnen Tiel West Maas en Waal Zaltbommel	<b>19. Alkmaar en omgeving</b>
<b>2. Delfzijl en omgeving</b>	<b>8. Zuidoost-Drenthe</b>			Akersloot Alkmaar Bergen Egmond Heerhugowaard Heiloo Langedijk Limmen Schermmer Schoorl
Appingedam Delfzijl Loppersum	Borger-Odoorn Coevorden Emmen		<b>17. Utrecht</b>	
<b>3. Overig Groningen</b>	<b>9. Zuidwest-Drenthe</b>		Abcoude Amerongen Amersfoort Baarn De Bilt Breukelen Bunnik Bunschoten Doorn Driebergen-Rijsenburg Eemnes Harmelen Houten IJsselstein Leersum Leusden Loenen Loosdrecht Lopik Maarn Maarssen Maartensdijk Montfoort Nieuwegein Oudewater Renswoude Rhenen De Ronde Venen Soest Utrecht Veenendaal Vleuten-De Meern Wijk bij Duurstede Woerden Woudenberg Zeist	<b>20. IJmond</b>
Bedum Ten Boer Eemsum Groningen Grootegast Haren Hoogezand-Sappemeer Leek De Marne Marum Slochteren Winsum Zuidhorn	Hoogeveen Meppel Westerveld De Wolden	<b>14. Achterhoek</b>		Beverwijk Castricum Heemskerk Uitgeest Velsen
<b>4. Noord-Friesland</b>	<b>10. Noord-Overijssel</b>	Aalten Bergh Borculo Brummen Dinxperlo Doetinchem Eibergen Gendringen Gorsel Groenlo Hengelo Hummelo en Keppel Lichtenvoorde Lochem Neede Ruurlo Steenderen Vorden Warnsveld Wehl Winterswijk Wisch Zelhem Zutphen		<b>21. Agglomeratie Haarlem</b>
Achtkarspelen Ameland het Bildt Boarnsterhim Dantumadeel Dongeradeel Ferwerderadiel Franekeradeel Harlingen Kollumerland c.a. Leeuwarden Leeuwarderadeel Littenseradiel Menaldumadeel Schiermonnikoog Terschelling Tytsjerksteradiel Vlieland	Avereest Brederwiede Dalfsen Genemuiden Gramsbergen Hardenberg Hasselt Heino IJsselham IJsselmuiden Kampen Nieuwleusen Ommen Staphorst Steenwijk Wijhe Zwartsluis Zwolle			Bennebroek Bloemendaal Haarlem Haarlemmerliede c.a. Heemstede Zandvoort
<b>5. Zuidwest-Friesland</b>	<b>11. Zuidwest-Overijssel</b>	<b>15. Arnhem/Nijmegen</b>		<b>22. Zaanstreek</b>
Bolsward Gaasterlân-Sleat Lemsterland Nijefurd Sneek Wûnseradiel Wymbritseradiel	Bathmen Deventer Holten Olst Raalte	Angerlo Arnhem Bemmel Beuningen Didam Doesburg Druten Duiven Elst Gendt Groesbeek Heteren Heumen Huissen Millingen aan de Rijn Nijmegen Renkum Rheden Rijnwaarden Rozendaal Ubbergen Valburg Westervoort Wijchen Zevenaar		Wormerland Zaanstad
<b>6. Zuidoost-Friesland</b>	<b>12. Twente</b>		<b>18. Kop van Noord-Holland</b>	<b>23. Amsterdam</b>
Heerenveen Ooststellingwerf Opsterland Skarsterlân Smallingerland Weststellingwerf	Almelo Ambt Delden Borne Denekamp Diepenheim Enschede Goor Haaksbergen Den Ham Hellendoorn Hengelo Losser Markelo Oldenzaal Ootmarsum Rijssen Stad Delden Tubbergen Vriezenveen Weerselo Wierden		Andijk Anna Paulowna Drechterland Enkhuizen Harenkarspel Den Helder Hoorn Medemblik Niedorp Noorder-Koggenland Obdam Opmeer Schagen Stede Broec Texel Venhuizen Wervershoof	Aalsmeer Amstelveen Amsterdam Beemster Diemen Edam-Volendam Graft-De Rijk Haarlemmermeer Landsmeer Oostzaan Ouder-Amstel Purmerend Uithoorn Waterland Zeevang
				<b>24. Het Gooi en Vechtstreek</b>
				Blaricum Bussum 's-Graveland Hilversum Huizen Laren Muiden Naarden Nederhorst den Berg Weesp

<b>25. Agglomeratie Leiden en Bollenstreek</b>	Dirksland Goedereede Heerjansdam Hellevoetsluis Korendijk Krimpen aan den IJssel Maassluis Middelharnis Nederlek Nieuwerkerk aan den IJssel Oostflakkee Oud-Beijerland Ouderkerk Ridderkerk Rotterdam Rozenburg Schiedam Spijkenisse Strijen Vlaardingen Westvoorne	Halderberge Moerdijk Oosterhout Roosendaal Rucphen Steenbergen Woensdrecht Zundert	Veldhoven Waalre
Alkemade Hillegom Katwijk Leiden Leiderdorp Lisse Noordwijk Noordwijkerhout Oegstgeest Rijnsburg Sassenheim Valkenburg Voorhout Voorschoten Warmond Zoeterwoude		<b>34. Midden-Noord-Brabant</b> Aalburg Alphen-Chaam Baarle-Nassau Dongen Gilze en Rijen Goirle Hilvarenbeek Loon op Zand Oisterwijk Tilburg Waalwijk Werkendam Woudrichem	<b>37.Noord-Limburg</b> Arcen en Velden Beesel Belfeld Bergen Broekhuizen Gennep Grubbenvorst Helden Horst Kessel Maasbree Meerlo-Wanssum Meijel Mook en Middelaar Sevenum Tegelen Venlo Venray
<b>26. Agglomeratie 's-Gravenhage</b>	<b>30. Zuidoost-Zuid-Holland</b> Alblasserdam Dordrecht Giessenlanden Gorinchem Graafstroom 's-Gravendeel Hardinxveld-Giessendam Hendrik-Ido-Ambacht Leerdam Liesveld Nieuw-Lekkerland Papendrecht Sliedrecht Vianen Zederik Zwijndrecht	<b>35. Noordoost-Noord-Brabant</b> Bernheze Boekel Boxmeer Boxtel Cuijk Grave Haaren 's-Hertogenbosch Heusden Landerd Lith Maasdonk Mill en Sint Hubert Oss Ravenstein Schijndel Sint Anthonis Sint-Michielsgestel Sint-Oedenrode Uden Veghel Vught	<b>38.Midden-Limburg</b> Ambt Montfort Echt Haelen Heel Heythuysen Hunsel Maasbracht Nederweert Roerdalen Roermond Roggel en Neer Swalmen Thorn Weert
's-Gravenhage Leidschendam Nootdorp Rijswijk Voorburg Wassenaar Zoetermeer			
<b>27.Delft en Westland</b>			
Delft 's-Gravenzande De Lier Maasland Monster Naaldwijk Pijnacker Schippluiden Wateringen			
<b>28.Oost-Zuid-Holland</b>	<b>31. Zeeuwsch-Vlaanderen</b> Axel Hontenisse Hulst Oostburg Sas van Gent Sluis-Aardenburg Terneuzen	<b>36. Zuidoost-Noord-Brabant</b> Asten Bergeijk Best Bladel Cranendonck Deurne Eersel Eindhoven Geldrop Gemert-Bakel Heeze-Leende Helmond Laarbeek Mierlo Nuenen c.a. Oirschot Reusel-De Mierden Someren Son en Breugel Valkenswaard	<b>39.Zuid-Limburg</b> Beek Born Brunssum Eijsden Geleen Gulpen-Wittem Heerlen Kerkrade Landgraaf Maastricht Margraten Meerssen Nuth Onderbanken Schinnen Simpelveld Sittard Stein Susteren Vaals Valkenburg aan de Geul Voerendaal
Ter Aar Alphen aan den Rijn Bergambacht Bodegraven Boskoop Gouda Jacobswoude Liemeer Moordrecht Nieuwkoop Reeuwijk Rijnwoude Schoonhoven Vlist Waddinxveen Zevenhuizen- Moerkapelle			
<b>29.Rijnmond</b>	<b>32.Overig Zeeland</b> Borsele Goes Kapelle Middelburg Noord-Beveland Reimerswaal Schouwen-Duiveland Tholen Veere Vlissingen		
Albrandswaard Barendrecht Bergschenhoek Berkel en Rodenrijs Bernisse Binnenmaas Bleiswijk Brielle Capelle aan den IJssel Cromstrijen	<b>33. West-Noord-Brabant</b> Bergen op Zoom Breda Drimmelen Etten-Leur Geertruidenberg		<b>40.Flevoland</b> Almere Dronten Lelystad Noordoostpolder Urk Zeewolde

(bron: CBS, 2000)



## Bijlage 2.3 Achtergrond regionale indelingen

Het Centraal Bureau voor de Statistiek (CBS) onderscheidt voor Nederland verschillende regionale indelingen. In het hoofdrapport wordt ingegaan op de Corop-indeling, die in het afstudeerwerk is gehanteerd. In deze bijlage worden de volgende regionale indelingen beknopt toegelicht:

- Corop-gebieden;
- Economisch-geografische gebieden;
- Samenwerkingsgebieden ingevolge de Wet gemeenschappelijke regelingen (*Wgr-gebieden*);
- Grootstedelijke agglomeraties;
- Stadsgewesten;
- Gemeentegroepen naar stedelijkheid.

### Bijlage 2.3.1 Corop-gebieden

De *Corop-indeling* is omstreeks 1970 ontworpen door de Coördinatie Commissie Regionaal Onderzoeksprogramma, waaraan de indeling haar naam dankt. Onderscheiden werden 40 gebieden, waaraan later de volgende drie subgebieden aan zijn toegevoegd.

Bij opzet van de indeling stond het nodale indelingsprincipe voorop. Daarbij gaat het met name om de functionele samenhang binnen een bepaalde regio tussen een of meer “centrale plaatsen” en het “omeland”. Centrale plaatsen zijn in dit verband stedelijke werk- en verzorgingscentra die als zodanig binnen een bepaalde regio als nodi (=knooppunten) van het maatschappelijk verkeer fungeren. Naar gelang van het aantal centrale plaatsen dat binnen iedere regio voorkomt zijn deze hetzij mono-nodaal (eenkernig) dan wel poly-nodaal (meerkernig). De leidraad bij de afbakening van de Corop-gebieden zijn 80 nodale gebieden geweest, die in het begin van de zestiger jaren is samengesteld op grond van de uitkomsten betreffende de intergemeentelijke forensenstromen van de Volkstelling van 31 mei 1960. Laatstbedoelde gegevens bieden namelijk een indicatie van de functionele relaties en daarmee van de onderlinge vervlechting tussen stedelijke centra en omliggende gemeenten op basis van het dagelijkse woon-werkverkeer.

Het nodale indelingsprincipe kon echter niet als enig criterium worden gehanteerd. Een beperking van het aantal gebieden tot 40 was noodzakelijk om kengetallen over toegevoegde waarde (naar bedrijfsklasse) en bruto-investeringen (naar type en bestemming) op dit niveau te kunnen presenteren.

Daarnaast hebben bij de reductie van het aantal gebiedseenheden ook overwegingen van overzichtelijkheid een rol gespeeld.

Verder was het wenselijk om in afwijking van het nodale indelingsprincipe in voorkomende gevallen de provinciegrens van een Corop-gebied aan te houden, temeer daar de provincies destijds als planregio's een belangrijke rol vervulden in de regionaal-economische ontwikkelingsmodellen van het Centraal Planbureau. Hiermee is de Corop-indeling een regionaal niveau tussen gemeenten en provincies in.

De Corop-indeling mag niet worden beschouwd als “standaard” indeling van Nederland. De Corop-indeling vervult slechts de functie van integratiekader van een aantal voor de ruimtelijke en regionaal-economische planning belangrijke basisstatistieken. Evenmin heeft de Corop-indeling blijvende geldigheid. Zo ligt het voor de hand dat, zodra er een nieuwe bestuurlijke indeling van Nederland van kracht zal worden, ook de Corop-indeling hierop zal moeten worden aangepast.

### **Bijlage 2.3.2 Economisch-geografische gebieden**

De indeling in *economisch-geografische gebieden* dateert van 1920. Onderscheiden werden toen 42 gebieden. De ontwerpers van de indeling gingen uit van het zonale principe om te komen tot homogene gebieden. Homogeen betekende in dit verband een overeenkomstig geografisch milieu en overeenkomst in de wijze waarop van dit milieu een productiegebied was gemaakt. Dat kon een agrarisch of een stedelijk-industrieel productiegebied zijn. Een en ander hield onder meer in dat van de 42 oorspronkelijke gebieden meer dan de helft gebaseerd was op de ook destijds geïntroduceerde landbouwgebieden.

In de loop der jaren is het aantal economisch-geografische gebieden bij diverse herzieningen toegenomen tot 129 in 1960. Voortgaande economische differentiatie en toename van het aantal steden en stedelijke agglomeraties maakten verdere opsplitsing noodzakelijk. Ook vonden opsplitsingen plaats doordat provinciegrenzen werden gevolgd. Vooral in westelijk en zuidelijk Nederland levert de indeling een fijnere regionale onderverdeling op dan die in Corop-gebieden. Sinds 1960 is de indeling op grond van overwegingen aangaande homogeniteit niet meer herzien.

### **Bijlage 2.3.3 Wgr-gebieden**

De indeling in 59 *Samenwerkingsgebieden* ingevolge de *Wet gemeenschappelijke regelingen* (Wgr) is een gevolg van de toenemende betekenis van intergemeentelijke samenwerking. Met behulp van deze wet wordt onder meer beoogd meer ordening in de patronen van gemeenschappelijke regelingen tot stand te brengen en de bundeling en integratie van dergelijke regelingen te bevorderen.

De indeling in samenwerkingsgebieden ingevolge de Wgr, ook wel Wgr-gebieden genoemd, speelt daarbij een belangrijke rol. De Wgr-gebieden worden door de onderscheiden provinciale staten vastgesteld. De Wgr zelf biedt de mogelijkheid samenwerkingsgebieden in te stellen die de grenzen van een provincie overschrijden. Voor een aantal Wgr-gebieden is dit thans niet het geval. De Wgr-indeling kan dus niet zonder meer worden beschouwd als een regionaal niveau tussen gemeenten en provincies in, dit in tegenstelling tot de indeling in Corop-gebieden en economisch-geografische gebieden.

### **Bijlage 2.3.4 Grootstedelijke agglomeraties**

Bij *grootstedelijke agglomeraties* gaat het om stedelijke gebieden in morfologische zin, dat wil zeggen om een aaneengesloten gebied met stedelijke bebouwing waarin de meeste menselijke activiteiten plaatsvinden, de meeste banen aanwezig zijn en de meeste openbare voorzieningen zijn gelokaliseerd. De grootstedelijke agglomeraties vormen de centra van de stadsgewesten. De gebieden met een stedelijke bebouwing zijn afgebakend aan de hand van de digitale kaart van het bodemgebruik naar de situatie van 1996. Welke van deze gebieden als centra van de stadsgewesten zijn beschouwd, is bepaald aan de hand van inwonertal (meer dan 100 duizend), aantal banen (meer dan 50 duizend) en potentieel aan regionale klanten (meer dan 150 duizend).

### **Bijlage 2.3.5 Het stadsgewest**

Het *stadsgewest* - het stedelijk gebied in functionele zin - bestaat uit een grootstedelijke agglomeratie en het omringende gebied met daarbinnen gelegen kleinere kernen (stadjes, dorpen, gehuchten) die met die agglomeratie door velerlei relaties met elkaar zijn verbonden. Die relaties hebben betrekking op het dagelijks verkeer tussen woon- en werkplaats, verhuizingen van huishoudens en bedrijven en het gebruik van stedelijke voorzieningen. Als zodanig kan het stadsgewest daarom worden beschouwd als een combinatie van een regionaal arbeid-, woning- en verzorgingsgebied. Voor de relaties tussen een grootstedelijke agglomeratie en het omringende gebied zijn twee criteria gehanteerd, namelijk:

- Forensenstromen op basis van gegevens over de woon- en werkgemeente afkomstig van de Enquête Beroepsbevolking over de gecombineerde verslagjaren 1995 t/m 1997;
- Verhuizingen binnen en tussen gemeenten afkomstig van de statistieken van de binnengemeentelijke verhuizingen en de binnenlandse migratie over de gecombineerde verslagjaren 1996 t/m 1997.

Voor ieder gehanteerd criterium is een niet-hiërarchische clusteranalyse uitgevoerd met een symmetrisch algoritme, waarna de uitkomsten daarvan in hun onderlinge samenhang zijn geanalyseerd. De uiteindelijke afbakening van de grootstedelijke agglomeraties en de stadsgewesten heeft plaatsgevonden in nauw overleg met de Commissie van Advies voor de Regionale Statistieken. In deze Commissie hebben de belangrijkste gebruikers van de regionale gegevens van het CBS (ministeries, gemeenten, planbureaus, wetenschap en bedrijfsleven) zitting.

### **Bijlage 2.3.6 Gemeentegroepen naar stedelijkheid**

De indeling van gemeenten naar *stedelijkheid* is gebaseerd op de adressendichtheid van de omgeving. Met de maatstaf adressendichtheid van de omgeving wordt beoogd de mate van concentratie van menselijke activiteiten (wonen, werken, voorzieningengebruik) weer te geven.

De mate van concentratie wordt daarbij beschouwd als kenmerk van de omgeving van een punt of locatie binnen een bepaald gebied. De mate van concentratie is geoperationaliseerd door het aantal adressen in de omgeving van een locatie; de omgeving door een gebied begrensd door een cirkel met een straal van 1 km en met als middelpunt een locatie, en de locatie door een adres. Bij de berekening van de adressendichtheden is gebruik gemaakt van het vierkantennet van 500x500 meter dat is gebaseerd op het coördinatensysteem van de Rijksdriehoekmeting.

De omgevingsadressendichtheid van een gemeente is gedefinieerd als het gemiddelde van de omgevingsadressendichtheden van alle afzonderlijke adressen binnen de gemeente. Deze maatstaf geeft dus aan hoe groot de adressendichtheid is in de omgeving van een gemiddeld adres in een gemeente. De gemeente wordt in dit verband dus meer beschouwd als een verzameling adressen waarop personen (kunnen) verblijven, dan als een bepaald gebied.

Voor de indeling naar stedelijkheid zijn afzonderlijke gemeenten gecategoriseerd in vijf klassen (zeer sterk stedelijk, sterk stedelijk, matig stedelijk, weinig stedelijk, niet stedelijk).



## **BIJLAGEN HOOFDSTUK 3**

### **METHODEN EN TECHNIEKEN VOOR SOCIAAL-ECONOMISCHE ANALYSE**



## Bijlage 3.1 Methoden en technieken

In tabel-Bijlage 3.1 zijn enkele traditionele modelmethoden- en technieken naar tijd- en ruimteaspect uiteengezet die in de praktijk gebruikt worden voor de analyse van achterliggende processen van ruimtelijke ontwikkelingen. Opgemerkt wordt dat statische modeltechnieken in principe ‘verdynamiseert’ kunnen worden toegepast, door het tijdselement hieraan toe te voegen. Daarnaast kunnen aan de modeltechnieken stochastische elementen worden toegevoegd.

Tabel-Bijlage 3.1 Sociaal-economische methoden en technieken naar tijd en ruimte

Tijd	Intraregionaal		Interregionaal		Ruimtelijke interactie
	voorraden	stromen	voorraden	stromen	
Statisch		Regionale Input-Output	Shift-Share; Lokatie-Quotiënt	Input-Output; Migratiestromen; Migratiesaldo; Entropie	Zwaartekracht; Push-Pull; Informatievelden; ‘Distance decay’; Logit; Netwerkanalyses
Tijdsklassen	Cohort-survival; Trendextrapolatie	Trendextrapolatie	Trendextrapolatie	Cohort-migratie; Trendextrapolatie	Account-based; Trendextrapolatie
Dynamisch	System Dynamics				Diffusie

(bron: Drewe, 1981; Krumme, 2001; Albrecht, 2001)

De meest aansprekende modelconcepten worden hieronder beknopt toegelicht.

## Bijlage 3.2 Zwaartekrachtmodellen

Het principe van de onderlinge aantrekkingskracht tussen twee onderlinge lichamen in de wet van Newton, werd door Ravenstein in 1885 geïntroduceerd ter analyse van migratie tussen twee gebieden. Deze analogie is nader uitgewerkt door met name Zipf (1946), Stewart (1947) en Wilson (1971) met betrekking tot de interactie tussen bevolkingscentra.

De zwaartekrachtmodellen kunnen worden gecategoriseerd volgens:

- A) ‘klassieke’ zwaartekrachtmodellen;
- B) zwaartekrachtmodellen met toegevoegde verklaringselementen;
- C) zwaartekrachtmodellen met beperkingen;
- D) zwaartekrachtmodellen met ‘intervening opportunities’.

Ad A) De ‘**klassieke**’ zwaartekrachtmodellen zijn reeds in de negentiende eeuw ontwikkeld door Ravenstein (1889) en later Wilson (1971). Hierin wordt verondersteld dat een stroom tussen een gebied  $i$  en gebied  $j$  omgekeerd evenredig is met de afstand tussen  $i$  en  $j$ , terwijl er een evenredig verband bestaat tussen een toestandsvariabele in  $i$  en een toestandsvariabele in  $j$ :

$$T_{ij} = c \frac{L_i L_j}{d_{ij}^\alpha} \quad (0.1)$$

Waarin:

- $T_{ij}$  = stroom tussen gebied  $i$  gebied  $j$ ;
- $L_i$  = toestandsvariabele van gebied  $i$ ;
- $L_j$  = toestandsvariabele van gebied  $j$ ;
- $d_{ij}$  = afstand tussen gebied  $i$  gebied  $j$ ;
- $\alpha$  = parameter behorende bij  $d_{ij}$ ;
- $c$  = constante.

De noemer van de zwaartekrachtformulering is in feite de weerstandsfunctie, welke refereert naar 'distance decay'. In de praktijk kan de weerstandsfunctie op verschillende wijze worden ingevuld. Een speciale vorm van de weerstandsfunctie is de 'logit'-formulering, welke is onderbouwd door de nuts-theorie. De logit-formulering wordt met name toegepast bij verkeerskundige analyses met betrekking tot vervoersmodekeuzes en routekeuzes.

De resultaten van klassieke gravitatiemodellen bleek in de praktijk beperkt, vanwege de veronderstelling dat de invloed van toestandsvariabelen en de afstand op de interactie bekend en voor elke relatie gelijk is. Dit leidde tot een overstap van beschrijvende naar verklarende benaderingen (zie B).

Ad B) Aan de zwaartekrachtmodellen werden vervolgens **verklaringselementen** toegevoegd. De stroom tussen gebieden is in deze modellen voor 'een helft' afhankelijk van de gravitatie van toestandsvariabelen en de afstand  $d_{ij}$ . Somermeyer (1970) voegt twee verklarende elementen aan het verschijnsel migratie toe: de relatieve attractie (o.b.v. gemiddeld inkomen per inkomenstrekker, werkloosheidspercentage, urbanisatiegraad, beschikbaar oppervlakte voor natuurrecreatie per inwoner, woningcomfort, percentage oude huizen) van een regio  $j$  tegenover een regio  $i$  en de sociale afstand (sociaal-culturele component) tussen de regio van vertrek en de regio van vestiging. Dit leidt tot de volgende formuleringen:

$$T_{ij} = \frac{1}{2} \frac{L_i L_j}{d_{ij}^\alpha} * \frac{\sum_f \{c_{of} + c_{if} (\bar{O}_{jf} - \bar{O}_{fi})\}}{(1 + \beta G_{ij})} \quad (0.2)$$

$$T_{ji} = \frac{1}{2} \frac{L_i L_j}{d_{ij}^\alpha} * \frac{\sum_f \{c_{of} - c_{if} (\bar{O}_{jf} - \bar{O}_{fi})\}}{(1 + \beta G_{ij})} \quad (0.3)$$

Met:

- $(\bar{O}_{jf} - \bar{O}_{fi})$  = relatieve attractie van gebied  $i$  tegenover gebied  $j$ ;
- $G_{ij}$  = sociale afstand tussen gebied  $i$  en gebied  $j$ ;
- $\alpha, \beta, c_{if}$  = te schatten parameters;
- $c_{of}$  = constante, waarvan verondersteld wordt dat hij verschillen van interindividuele preferentie of waardering weergeeft met betrekking tot de attractie  $\bar{O}_f$ .

De relatieve attractie  $(\bar{O}_{jf} - \bar{O}_{fi})$  kan door meerdere factoren worden gemeten. Bij migratie bijvoorbeeld het gemiddelde inkomen per inkomenstrekker, het werkloosheidspercentage, de urbanisatiegraad, de oppervlakte beschikbaar voor natuurrecreatie per inwoner, het woningcomfort en het percentage oude huizen.

Op deze manier kunnen sociaal-economische processen op geaggregeerd niveau beschreven worden.

De methode is vooral theoretisch interessant, en is vanwege een gebrek aan databeschikbaarheid niet eenvoudig te operationaliseren.

Ad C) Overige *ruimtelijke interactiemodellen* zijn verfijningen van de klassieke zwaartekrachtmodellen. Rekening houdend met het in bepaalde gevallen optreden van **beperkingen**, onderscheiden Cordey-Hayes en Wilson (Drewe e.a. 1981) vier typen ruimtelijke interactiemodellen:

- a) Modellen zonder beperkingen ('unconstrained');
- b) Modellen met productiebeperking ('single production constrained');



- c) Modellen met attractiebeperking ('*single attraction constrained*');  
 d) Modellen met productie- " én attractiebeperking ('*double constrained*').

ad a) Modellen zonder beperkingen hebben dezelfde vorm als de klassieke zwaartekrachtmodellen:

$$T_{ij} = c \frac{O_i O_j}{d_{ij}^\alpha} \quad (0.4)$$

Met:

- $O_i$  = productievariabelen (in termen van verplaatsingen) van oorsprong  $i$ ;  
 $O_j$  = attractievariabele(n) van bestemming  $j$ ;  
 $O_i O_j$  = 'natuurlijke geneigdheid' van een regio  $i$  ( $j$ ) om een stroom voort te brengen (aan te trekken).

De productie- en attractievariabelen kunnen worden afgeleid van bevolking, aantal banen, gebiedsoppervlakte en dergelijke. Het totaal aan vertrek (productie) en vestiging (attractie) zijn onbekend en worden endogeen bepaald. Elke flow wordt enkel bepaald op basis van potenties van regio  $i$  en  $j$  en de weerstand hiertussen, zonder rekening te houden met beperkingen. De modelstructuur is hierdoor niet altijd realistisch.

ad b) Modellen met *productiebeperking* (herkomst bekend, bestemming wordt afgeleid) hebben de vorm:

$$T_{ij} = T_{\text{vertrek},i} \frac{O_j F_{ij}}{\sum_j (O_j F_{ij})} = x_i T_{\text{vertrek},i} O_j F_{ij} \quad (0.5)$$

$$x_i = \frac{1}{\sum_j (O_j F_{ij})} \quad (0.6)$$

$$F_{ij} = \frac{1}{d_{ij}^\alpha} \quad (0.7)$$

$$T_{\text{vertrek},i} = \sum_j T_{ij} \quad (0.8)$$

Met:

- $T_{\text{vertrek},i}$  = totaal vertrek uit regio  $i$  (exogeen);  
 $F_{ij}$  = bereikbaarheid van regio  $j$  vanuit regio  $i$ ;  
 $x_i$  = schaalfactor.

Het totaal aan vertrek  $T_{\text{vertrek},i}$  uit regio  $i$ , in de formulering exogeen verondersteld, wordt *proportioneel* over de bestemmingen  $j$  verdeeld op basis van *relatieve* bereikbaarheid en *relatieve* attractiviteit van de bestemming  $j$ .

ad c) Modellen met *attractiebeperking* (bestemming bekend, herkomst wordt afgeleid) hebben de vorm:

$$T_{ij} = T_{\text{vestiging},j} \frac{O_i F_{ij}}{\sum_i (O_i F_{ij})} = y_j T_{\text{vestiging},j} O_i F_{ij} \quad (0.9)$$

$$y_j = \frac{1}{\sum_i (O_i F_{ij})} \quad (0.10)$$

$$F_{ij} = \frac{1}{d_{ij}^\alpha} \quad (0.11)$$

$$T_{\text{vestiging},j} = \sum_i T_{ij} \quad (0.12)$$

Met:

$T_{\text{vestiging},j}$  = totaal vestiging in regio  $j$  (exogeen);  
 $y_j$  = schaalfactor.

Het totaal aan vestiging per regio ( $F_{imm}$ ) is exogeen verondersteld. Het totaal aan vestigingen in regio  $j$  uit de regio's  $i$  wordt *proportioneel* afgeleid op basis van *relatieve bereikbaarheid* en *relatieve attractiviteit* van de herkomstregio's  $i$ .

ad d) Modellen met *productie- en attractiebeperking* hebben de vorm:

$$T_{ij} = v_i w_j T_{\text{vertrek},i} T_{\text{vestiging},j} F_{ij} \quad (0.13)$$

$$v_i = \frac{1}{\sum_j (O_j F_{ij})} \quad (0.14)$$

$$w_j = \frac{1}{\sum_i (O_i F_{ij})} \quad (0.15)$$

$$F_{ij} = \frac{1}{d_{ij}^\alpha} \quad (0.16)$$

$$T_{\text{vertrek},i} = \sum_j T_{ij} \quad (0.17)$$

$$T_{\text{vestiging},j} = \sum_i T_{ij} \quad (0.18)$$

Met:

$v_i, w_j$  = schaalfactoren.

In dit model zijn zowel het totaal aan vestiging als het totaal aan vertrek per regio exogeen bekend verondersteld. In tegenstelling tot de onbeperkte en de enkel beperkte modellen, kan het dubbel beperkte model niet direct worden berekend. De schaalfactoren worden iteratief bepaald.

Donovan (1971) heeft als een van de eersten interregionale migratiestromen berekend met een zwaartekrachtmodel met dubbele beperking;  $F_{imm}$  en  $F_{em}$  zijn daarbij geschat door middel van een regressieanalyse.

Ad D) Stouffer (1940) stelt dat de stromen van  $i$  naar  $j$  mede afhankelijk is van de aantrekkelijkheid van tussenliggende gebieden ('*intervening opportunities*'). De analogie refereert naar **informatievelden**. De formulering lijkt veel op het klassieke zwaartekrachtmodel, alleen de afstandsterm is vervangen door het (gecumuleerde) aantal 'tussenliggende attracties'. Qua werking, is het model vergelijkbaar met de schaalfactoren in ruimtelijke interactiemodellen. Wilson (1967) formuleert 'intervening opportunities' als volgt:

$$T_{ij_\mu} = c_i \frac{T_{em}}{e^{PO_{j_\mu^{-1}}} - e^{PO_{j_\mu}}} \quad (0.19)$$

Met:

- $M_{ij\mu}$  = stroom vanuit regio  $i$  naar de  $\mu^{\text{de}}$  bestemming ( $j_\mu$ ), gerekend vanaf  $i$ ;
- $P$  = kans dat de behoeften van een stromer (bijv. immigrant) uit  $i$  worden bevredigd (dan is  $j_\mu=1$ );
- $O_{j\mu}$  = het aantal attracties tot en met regio  $j$  vanaf regio  $i$ ;
- $j_\mu$  = de  $\mu^{\text{ste}}$  bestemming in een bepaalde rangorde vanaf  $i$ ;
- $c_i$  = constante.

### Bijlage 3.3 Push-pull modellen

De **push-pull modellen** maken gebruik van de analogie van de klassieke zwaartekrachtmodellen en regressievergelijkingen. De toestandsvariabelen (of massa's) zijn economische pushes en pulls die stromen tussen regio's vertegenwoordigen; de afstand wordt nog steeds gehanteerd. Alle parameters worden berekend i.p.v. gedetermineerd.

Het eerste model is ontwikkeld door Lowry (1966). Lowry stelt dat de omvang van de migratie van regio  $i$  naar regio  $j$  evenredig is met de omvang van de niet-agrarische beroepsbevolking in  $i$  en  $j$ , de werkeloosheid in  $i$ , en het loonniveau in  $j$  en dat de omvang van de migratiestroom omgekeerd-evenredig is met de werkeloosheid in  $j$ , het loonniveau in  $i$  en tenslotte de afstand (hemelsbreed) tussen regio van vertrek en regio van vestiging. In formulevorm is het model van Lowry (1966):

$$T_{ij} = \alpha_0 \frac{U_i^{\alpha_1} W_j^{\alpha_3} L_i^{\alpha_5} L_j^{\alpha_6}}{U_j^{\alpha_2} W_i^{\alpha_4} d_{ij}^{\alpha_7}} \quad (0.20)$$

Waarin:

- $T_{ij}$  = stroom (migratie) vanuit regio  $i$  naar regio  $j$ ;
- $L_i$  = niet-agrarische beroepsbevolking in regio  $i$ ;
- $L_j$  = niet-agrarische beroepsbevolking in regio  $j$ ;
- $U_i$  = werkeloosheid in regio  $i$ ;
- $U_j$  = werkeloosheid in regio  $j$ ;
- $W_i$  = loonniveau in regio  $i$ ;
- $W_j$  = loonniveau in regio  $j$ ;
- $d_{ij}$  = hemelsbrede afstand tussen regio  $i$  en regio  $j$ ;
- $\alpha_0$  = constante term;
- $\alpha_7$  = afstandsgevoeligheid;
- $\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6$ , zijn regressiecoëfficiënten.

Het originele Lowry-model kan verfijnd en aangepast worden. Lowry toont aan dat opsplitsing van beroepsbevolking naar 'burgerlijk' en 'militair' een toegenomen verklaring geeft voor migratiestromen.

Rogers (1968) bereikte verbetering in resultaat door  $d_{ij}$  te interpreteren als de afstand over de weg. Rogers weet geen verklaring voor het 'pull-effect' van de werkeloosheid in de regio van vestiging, en past het Lowry-model aan. Het *Rogers-model* luidt als volgt:

$$T_{ij} = \beta_0 \frac{W_j^{\beta_1} L_i^{\beta_3} L_j^{\beta_4}}{W_i^{\beta_2} d_{ij}^{\beta_5}} \quad (0.21)$$

In Engeland (Masser 1970) en VS (Californie) zijn met het Lowry-model en Rogers-model algemeen goede resultaten geboekt. Verbetering in resultaat kan bereikt worden door entiteiten in de formulering te desaggregeren (bijvoorbeeld geslacht).

### Bijlage 3.4 Input-Output modellen

Een **Input-Output-model** (*I/O-model*) geeft een sectorale, economische toestandsbeschrijving en analyseert de interregionale relaties d.m.v. flows van goederen en diensten tussen economische sectoren. De I/O-methode is gebaseerd op de fundamentele veronderstelling dat voor de productie van goederen ('output') andere goederen nodig zijn ('input'). Inputs zijn bijvoorbeeld grondstoffen, halffabrikaten, of diensten van huishoudens (arbeid) of de overheid.

Soms worden de transacties door zwaartekrachtmodellen bepaald. Vaak zijn het top-down modellen, waarbij de nationale vraag d.m.v. exogene verhoudingen over regio's verdeeld wordt (Courbis, 1984).

De I/O-methode is dubbel-boekhoudkundige benadering waarbij van matrixformuleringen gebruik wordt gemaakt. De formulering van de I/O-methode is als volgt:

$$x_i = \sum_j x_{ij} + \sum_j y_{ij} \quad (0.22)$$

$$x_i = \sum_i x_{ij} + \sum_i v_{ij} \quad (0.23)$$

Waarin:

- $x_i$  = totale input regio  $i$  = totale output regio  $i$ ;
- $x_{ij}$  = transactiewaarde van goederen en diensten dat van regio  $i$  aan regio  $j$  verkocht is;
- $y_{ij}$  = vraag van regio  $j$  naar input (huishoudens, overheid, investeringen en export) uit regio  $i$ ;
- $v_{ij}$  = primaire input (zoals lonen, directe belastingen, indirecte belastingen minus subsidies, afschrijvingen, import) van regio  $i$  noodzakelijk voor regio  $j$ .

De I/O-methode kan worden vertaald in een model dat veranderingen in de economie analyseert (I/O-analyse). Hiervoor worden inputcoëfficiënten (technische coëfficiënten) constant verondersteld. Dit leidt tot de volgende formulering:

$$a_{ij} = \frac{x_{ij}}{x_j} \quad (0.24)$$

$$x_i = \sum_j (a_{ij} x_j) + \sum_j y_{ij} \quad (0.25)$$

Met:

- $a_{ij}$  = constante technische coëfficiënt;
- $x_j$  = totale input regio  $j$  = totale output regio  $j$ .

Dit systeem kan in matrixvorm beschreven:

$$\bar{x} = A\bar{x} + \bar{y} \quad (0.26)$$

Waarin:

- $\bar{x}$  = set van  $x_i$ ;
- $\bar{y}$  = set van  $y_{ij}$ ;
- A = set van  $a_{ij}$ .

Na enkele simpele matrixberekeningen ontstaat het *I/O-model*:

$$\bar{x} = (I - A)^{-1} y \quad (0.27)$$

Met:

$I$  = identiteitsmatrix, met zelfde afmeting als  $A$ ;  
 $(I-A)^{-1}$  = zgn. Leontief inverse.

Het *I/O-model* in een dynamische formulering is als volgt:

$$d\bar{x} = (I - A)^{-1} d\bar{y} \quad (0.28)$$

Een exogene verandering in de eindvraag ( $dy$ ) geeft endogeen een verandering in de totale output ( $dx$ ).

*I/O-modellen* beschrijven en voorspellen een economisch evenwicht, maar geven geen inzicht in de processen die tot het evenwicht leiden. Deze traditionele modellen modelleren het keuzegedrag van bedrijven niet expliciet, waardoor ruimtelijk-economisch beleid niet goed gesimuleerd en geëvalueerd kan worden.

### Bijlage 3.5 Shift-Share

De **Shift-Share** methode is in feite geen model, maar een rekenkundige uitsplitsing, ter analyse van veranderingen in lokale economische structuur in relatie tot een referentie-economie. In deze methodiek wordt de nationaal/regionaal economische groei (op basis van de nationale/regionale bedrijfstakontwikkeling) met behulp van de lokale bedrijfssectorsamenstellingen naar de lokaal economische groei vertaald. De lokaal economische groei is de 'share'. De mate waarin locaties anders groeien dan die door de nationale/regionale bedrijfstakontwikkeling is bepaald, is de 'shift'.

Voor twee jaren dienen voor zowel de lokale als referentie-economie gegevens beschikbaar te zijn. De Shift-Share analyse splitst vervolgens de lokale economische performance uit naar drie componenten: (1) een economische groeicomponent, (2) een proportionele shift-component en (3) een differentiële shift-component. Op basis van deze componenten wordt de 'share' bepaald.

De economische groeicomponent geeft de werkgelegenheidsverandering in de referentie-economie weer (Arizona State University, 2000):

$$egc = \frac{E_T^{(t+x)} - E_T'}{E_T'} \quad (0.29)$$

Waarin:

$egc$  = economische groeicomponent;  
 $E_T^{(t+x)}$  = werkgelegenheid in referentie-economie op tijdstip  $(t+x)$ ;  
 $E_T'$  = werkgelegenheid in referentie-economie op tijdstip  $t$ .

De proportionele shift-component geeft de relatieve ontwikkeling (groei of krimp) van de bedrijfssector in de referentie-economie weer in vergelijking tot de ontwikkeling van de totale economie.

$$psc = \frac{E_i^{(t+x)}}{E_i'} - \frac{E_T^{(t+x)}}{E_T'} \quad (0.30)$$

Met:

$psc$  = proportionele shift-component;

$E_i^{(t+x)}$  = werkgelegenheid in bedrijfstak  $i$  van referentie-economie op tijdstip  $(t+x)$ ;

$E_i'$  = werkgelegenheid in bedrijfstak  $i$  van referentie-economie op tijdstip  $t$ .

De differentiële shift-component geeft de performance van een lokale bedrijfssector in verhouding tot de performance van diezelfde bedrijfssector in de referentie-economie weer.

$$dsc = \frac{e_i^{(t+x)}}{e_i'} - \frac{E_i^{(t+x)}}{E_i'} \quad (0.31)$$

Met:

$dsc$  = differentiële shift-component;

$e_i^{(t+x)}$  = werkgelegenheid in bedrijfstak  $i$  van lokale economie op tijdstip  $(t+x)$ ;

$e_i'$  = werkgelegenheid in bedrijfstak  $i$  van lokale economie op tijdstip  $t$ .

$$share = egc + psc + dsc \quad (0.32)$$

Met:

$share$  = lokale werkgelegenheidsontwikkeling in bedrijfstak  $i$  tussen  $(t+x)$  en  $t$ .

Aan de methodiek kunnen verklarende elementen worden toegevoegd, door de shift te verklaren aan de hand van locatiefactoren ('te verklaren variabelen', die een weergave zijn van alle bedrijfsexterne factoren die het bedrijfsfunctioneren potentieel beïnvloeden en waardoor op referentieniveau ruimtelijke verschillende bestaan in intensiteit en/of kwaliteit bestaan). De methodiek wordt dan een 'verklarende Shift-Share' analyse genoemd. Net als I/O-modellen blijven Shift-Share analyses in causaal opzicht zwak.

### Bijlage 3.6 Lokatie-Quotiënt

In deze methodiek wordt de lokale economie vergeleken met een overkoepelende economie (regionaal of nationaal). De **lokatie-quotiënt** wordt gedefinieerd als de verhouding tussen de 'share' van een bedrijfssector in de lokale economie en de 'share' van die bedrijfssector in de overkoepelende economie (Arizona State University, 2000).

$$LQ_i = \frac{\frac{e_i^t}{E_T^t}}{\frac{E_i^t}{E_T^t}} \quad (0.33)$$

Waarin:

$LQ_i$  = lokatie-quotiënt;

$e_i^t$  = lokale werkgelegenheid in bedrijfssector  $i$  in jaar  $t$ ;

$e_t^i$  = totale werkgelegenheid in jaar  $t$ ;

$E_i^t$  = nationale/regionale werkgelegenheid in bedrijfssector  $i$  in jaar  $t$ ;

$E_t^t$  = nationale/regionale werkgelegenheid in jaar  $t$ .

Bedrijfssectoren met  $LQ=1$  hebben een lokale werkgelegenheids-‘share’ die gelijk is aan de ‘share’ in de overkoepelende referentie economie. Er wordt dan verondersteld dat de lokale productie van de bedrijfssector ongeveer voldoende is om aan de lokale vraag te voldoen, waarbij wordt verondersteld dat de bedrijfssector geen basis-werkgelegenheid biedt.

Bedrijfssectoren met  $LQ<1$  hebben een lokale werkgelegenheids-‘share’ die kleiner is dan de ‘share’ in de overkoepelende referentie economie. Er wordt dan verondersteld dat de lokale productie van de bedrijfssector onvoldoende is om aan de lokale vraag te voldoen, waardoor import noodzakelijk is.

Bedrijfssectoren met  $LQ>1$  hebben een lokale werkgelegenheids-‘share’ die groter is dan de ‘share’ in de overkoepelende referentie economie. Er wordt dan verondersteld dat de lokale productie van de bedrijfssector gespecialiseerd is en de lokale vraag overtreft, waardoor de bedrijfssector exportgericht is.

### Bijlage 3.7 Migratiestroom-modellen

Naar het idee van de intervenie mogelijkheden wordt de migratiestroom van  $i$  naar  $j$  gereduceerd door de migratiestromen vanuit  $i$  naar de tussenliggende gebieden. Bij push-pull modellen hangt de migratiestroom tussen regio  $i$  en regio  $j$  allen af van de structurele kenmerken van beide regio’s, zonder de structuur van een derde regio hier bij te betrekken. De **migratiestroom- of ‘migration-rate’-modellen** (*‘flow diversion’*) combineren de analogie van de intervenie mogelijkheden met die van de push-pull modellen.

De migratie van regio  $i$  en regio  $j$  hangt in de eerste plaats af van structurele kenmerken van  $i$  en  $j$ , maar ook samen met de omvang van migratiestromen naar andere regio’s. Glesjer en Dramais (1969) geven de volgende formulering:

$$T_{ij} = \sum_k (x_{ij}^k \beta_k) + \alpha \log \sum_l T_{il} \quad (0.34)$$

Waarin:

$T_{ij}$  = stroom (migratie) vanuit regio  $i$  naar regio  $j$ ;

$x_{ij}$  = structureel kenmerk tussen  $i$  en  $j$ ;

$k$  = aantal kenmerken;

$\beta_k$  = elasticiteitscoëfficiënten behorende bij kenmerk  $k$  van regio  $i$  en regio  $j$ ;

$l$  = aantal regio’s;

$T_{il}$  = omvang migratiestromen naar andere regio’s dan  $j$ ;

$\alpha$  = regressiecoëfficiënt.

### Bijlage 3.8 Migratiesaldo-modellen

Bij **migratiesaldo-modellen** wordt niet het detail van uitgaande en inkomende migratie onderscheiden, maar enkel het migratiesaldo. De meeste van dergelijke modellen zijn de *primaire economische benaderingen*.

De primair economische benaderingen veronderstellen dat er migratie (van beroepsbevolking) van een regio  $i$  naar een regio  $j$  plaatsvindt zolang het gemiddelde loon-niveau in  $j$  boven dat van  $i$  uitkomt.

$$T_{netto} = \alpha(W - \bar{W}) \quad (0.35)$$

Waarin:

$$\begin{aligned} T_{netto} &= \text{netto-stroom (netto-migratie) in regio } i; \\ W &= \text{loonniveau in regio } i; \\ \bar{W} &= \text{gemiddelde loonniveau van overige regio's.} \end{aligned}$$

Veronderstellingen hierbij zijn:

- Migranten streven naar maximalisatie van inkomen;
- De informatie m.b.t. werkgelegenheid is perfect;
- Migranten zijn homogeen wat betreft hun kwalificatie en preferentie;
- Er bestaan geen sociale en economische migratiebarrières.

Onder deze omstandigheden bevindt zich iedere regionale arbeidsmarkt altijd in evenwicht. Reële loonverschillen zorgen voor een aanbod dat even groot is als de vraag, zodat er geen werkloosheid is. Additionele variabelen worden geïntroduceerd om met afwijkingen tussen de realiteit en het model van perfecte competitie af te houden. Muth (1969) gaat ervan uit dat werkgelegenheidsgroei niet alleen de netto-migratie beïnvloedt, maar dat tevens de netto-migratie de groei van de werkgelegenheid beïnvloedt:

$$T_{netto} = \alpha_1 \Delta E + \alpha_2 Y + \alpha_0 \quad (0.36)$$

$$\Delta E = \beta_1 M + \beta_0 \quad (0.37)$$

Met:

$$\begin{aligned} \Delta E &= \text{groei werkgelegenheid;} \\ Y &= \text{inkomen;} \\ \alpha_0, \beta_0 &= \text{constante termen;} \\ \alpha_1, \alpha_2, \beta_1 &= \text{zijn regressiecoëfficiënten.} \end{aligned}$$

### Bijlage 3.9 Cohort-Survival

De **cohortenmethode** (Cannon 1895) is de meest gebruikte methoden ten aanzien van het prognosticeren van populaties (bijvoorbeeld bevolking of bedrijven) en geeft gedetailleerde informatie over de omvang en samenstelling (*demografie*) van de populatie. Bij de methode wordt de populatie gesegmenteerd naar leeftijdsgroepen. Dit zijn de cohorten. Op basis van empirische gegevens wordt voor elk cohort de overlevingskans bepaald. Dit is de kans dat een persoon de tijdstap overleeft en daarmee in het volgende cohort terecht komt.

De cohortberekening is vrij nauwkeurig, doordat per cohort de veranderingen worden bepaald. Dit methode is hierdoor tevens erg informatief. Doordat de omvang en kenmerken van elke leeftijdsklasse bekend zijn, is veel materiaal beschikbaar voor verdere berekeningen. Veel processen zijn leeftijds- en kenmerkafhankelijk. Doordat de waarde van cohort-modellen groot is, wordt deze methodiek in de praktijk veelvuldig gebruikt.



## **BIJLAGEN HOOFDSTUK 4**

### **HANDLEIDING ARRAY'S IN STELLA**



## Bijlage 4.1 Traditionele toepassingen van array's in Stella

Met array's kunnen in Stella complexe systemen op een sterk vereenvoudigde wijze worden weergegeven. Indien zeer complexe systemen in beeld moeten worden gebracht, is het gebruik van array's in methodisch opzicht van cruciaal belang. Vaak betekent dit dat er creatief met het instrument moet worden omgegaan.

Deze bijlage is een handleiding voor het gebruik van array's in Stella, waarbij de **traditionele gebruikswijze** van het array-instrument aan bod komt. Niet in deze bijlage, maar in bijlage 4.2 wordt aangegeven hoe het array-instrument in zeer complexe systemen in een **creatieve benadering** van grote waarde kan zijn.

Eerst wordt het begrip array in paragraaf bijlage 4.1.1 gedefinieerd. In bijlage 4.1.2 wordt het idee geïntroduceerd aan de hand waarvan de traditionele toepassing van de array-functie binnen Stella wordt toegelicht. Het voorbeeld dat een toepassing van de enkelvoudige array-functie kan illustreren is beschreven in paragraaf 4.1.3. De manier waarop de enkelvoudige array gebruikt wordt in dit voorbeeld is beschreven in bijlage 4.1.4. In bijlage 4.1.5 wordt het gebruikte voorbeeld uitgebreid en wordt uitgelegd hoe de dubbele array hierbij toegepast kan worden.

### Bijlage 4.1.1 Begripsdefiniëring array

Een *array* is een verzameling van een vast aantal elementen van hetzelfde type, die daarin geïndiceerd zijn. Een array kan voorgesteld worden als een soort *kaartenbak*: ook hierin zijn een aantal elementen van dezelfde structuur ('kaarten') opgeslagen. Een array kan meerdimensionaal zijn. Stella kent *enkelvoudige* (eendimensionale) en *dubbele* (tweedimensionale) array's.

Een **enkelvoudige array** kan *op papier* beschouwd worden als een **vector**. Een **dubbele array** stelt *op papier* een **matrix** voor.

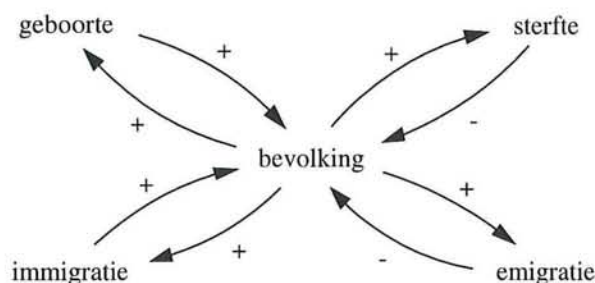
Ieder element in een array is toegankelijk via zijn indexnummer. De indicering van een enkelvoudige array is *Array[rij]*. De indicering van de dubbele array is *Array[rij, kolom]*.

### Bijlage 4.1.2 Het idee: een regionaal bevolkingsmodel

De methodiek van de array wordt uitgelegd aan de hand van een simpel, sterk vereenvoudigd bevolkingsmodel. Volgens de systeemkunde dient eerst de causale achtergrond van de bevolkingsontwikkeling in beeld te worden gebracht, waarna de modellering van structuur in Stella kan worden gemaakt.

#### Definiëring van de causaliteit

De grootte van de bevolking kan zich ontwikkelen door geboorte, sterfte en/of migratie van personen. Stel dat in eerste instantie *Nederland* als schaalniveau wordt bekeken. Het causale relatiediagram van de Nederlandse bevolkingsontwikkeling is weergegeven in figuur-Bijlage 4.1. Figuur-Bijlage 4.1 laat zien dat door geboorte en immigratie de Nederlandse bevolking toeneemt.

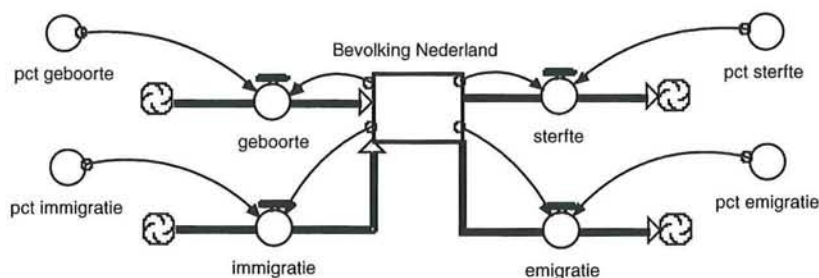


Figuur-Bijlage 4.1 Causaal relatiediagram Nationale bevolkingsontwikkeling

Door sterfte en emigratie neemt de bevolking af. Verder worden ook proportionele verbanden aangenomen tussen de grootte van de bevolking en geboorte/sterfte/migratie. Het lijkt aannemelijk dat in een twee keer zo grote bevolking ongeveer twee keer zoveel mensen geboren worden c.q. sterven. Dezelfde redenatie kan worden toegepast bij migratie. Uit een grotere bevolking zullen meer mensen emigreren. Het proportionele verband tussen immigratie en de grootte van de bevolking kan verklaard worden door het sociale karakter van migratie. Via sociaal netwerken is het mogelijk dat Nederlandse inwoners mensen uit het buitenland op de hoogte brengen van bijvoorbeeld gunstige woon- en arbeidsmarktomsstandigheden. Vanuit een grotere bevolkingspopulatie worden relatief meer personen uit het buitenland op de hoogte gebracht. Er kan hierbij gedacht worden aan familiereünies. Het lijkt dan ook aannemelijk dat een grotere bevolkingsomvang per saldo meer immigranten aantrekt.

### *De vertaling naar een Stella-modellering*

De bevolking kan geïnterpreteerd worden als een voorraad, welke accumuleert door geboorte, immigratie, sterfte en emigratie. Het bijbehorende Stella-model van het ontwikkelingsproces van de Nederlandse bevolking is weergegeven in figuur-Bijlage 4.2.



*Figuur-Bijlage 4.2 Modellering Nederlandse bevolkingsontwikkeling*

Het proportionele verband tussen de bevolking en ontwikkelingen hierin wordt steeds weergegeven met behulp van een percentage. De grootte van geboorte, sterfte en migratie wordt verkregen door vermenigvuldiging van de grootte van de bevolking met dit percentage.

### **Bijlage 4.1.3 Het voorbeeld: een multiregionaal bevolkingsmodel**

Het is veel informatiever om meerdere, kleinere ruimtelijke eenheden in kaart te brengen. Nederland kan bijvoorbeeld opgedeeld worden in 40 regio's -de zogenaamde Corop-regio's- waarvan relatief veel statistiek bestaat.

Er bestaan regionale verschillen in geboorte-, sterfte- en migratiepercentages. In de praktijk zijn sommige regio's van nature aantrekkelijker voor immigranten dan andere regio's. Een immigrant van de andere kant van de wereld is waarschijnlijk beter bekend met Amsterdam dan met Groningen en zal zich dus eerder in Amsterdam vestigen. Daarnaast leiden regionale verschillen in de samenstelling van de bevolking (zoals leeftijdsopbouw, geslacht, etnische achtergrond) tot regionaal verschillende percentages met betrekking tot geboorte, sterfte en migratie.

Indien nog steeds Nederland wordt bestudeerd, maar nu via 40 regio's, dient elke regio in modellering te worden gebracht. Sommatie van alle 40 regio's leveren de Nederlandse waarden op. Een dergelijk model is veel informatiever dan een losstaand Nederlands model, aangezien op een veel lager schaalniveau informatie in beeld wordt gebracht. Dit biedt tevens betere mogelijkheden om relevante processen in beeld te brengen.

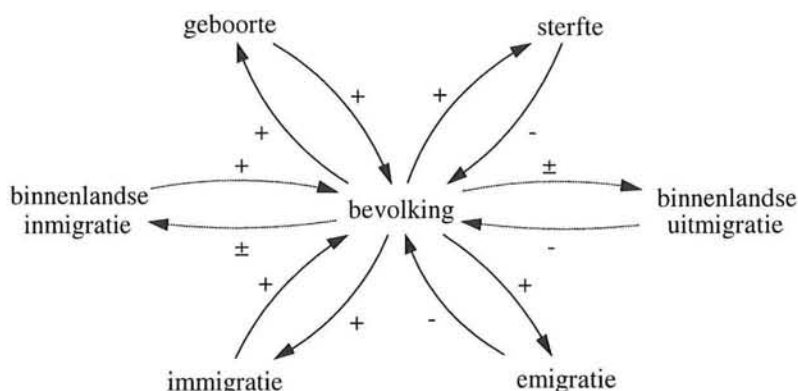
Er doen zich nu twee essentiële veranderingen in het voorbeeld van het bevolkingsmodel voor:

- 1 het analytisch schaalniveau van de bevolkingsontwikkeling verandert;
- 2 de causaliteit van de bevolkingsontwikkeling verandert.

### Aanpassen van het causaal relatiediagram

Nog steeds wordt Nederland bekeken, maar nu via meerdere regio's. Nederland is nog steeds het beschouwde *stelsel*, maar dit is nu opgebouwd uit meerdere *componenten* (40 regio's). De verandering van schaalniveau waarop de bevolking in beeld wordt gebracht, heeft ook gevolgen voor de causale weergave van de bevolkingsontwikkeling. Op regionaal niveau speelt naast natuurlijke aanwas (geboorte en sterfte) en buitenlandse migratie (immigratie en emigratie) ook binnenlandse migratie een rol. Sterker nog, de binnenlandse migratie is verreweg het meest aanpassende marktmechanisme op veranderingen in regionale marktomstandigheden. Indien in Amsterdam veel woningen worden gebouwd leidt dit eerder tot meer immigranten die ook uit Nederland komen en op de hoogte zijn van deze ontwikkeling, dan tot een toename van het aantal immigranten afkomstig van de andere kant van de wereld en niets van die woningbouw afweten.

De aanpassing van schaalniveau leidt dus tot een aanpassing van het causale relatiediagram voor het weergegeven van bevolkingsontwikkeling. De regionale bevolkingsontwikkeling van een regio is weergegeven in figuur-Bijlage 4.3.

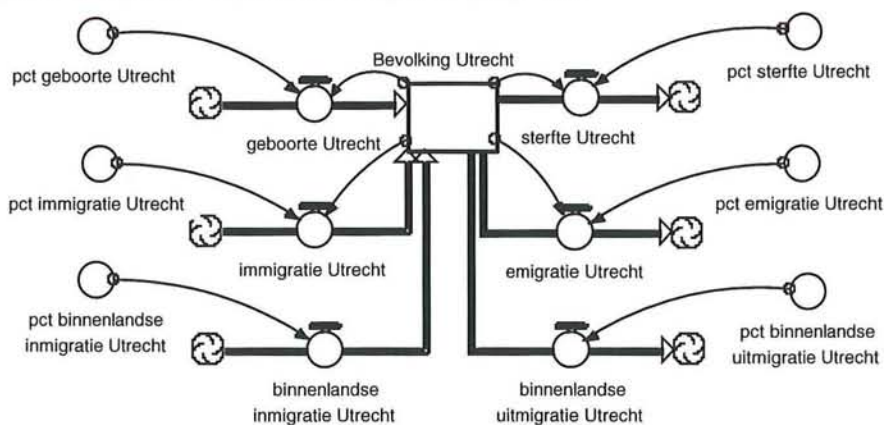


Figuur-Bijlage 4.3 Causaal relatiediagram regionale bevolkingsontwikkeling

Veel meer dan bij buitenlandse migratie, spelen bij binnenlandse migratie de regionale marktomstandigheden een rol. Dit verklaart het ongedefinieerde teken van proportionaliteit tussen de grootte van de bevolking en de binnenlandse migratie. Door binnenlandse immigratie neemt de regionale bevolking toe, door binnenlandse uitmigratie (verhuizing naar een andere Nederlandse regio) neemt de bevolking in de regio van beschouwing af.

### Aanpassen van de Stella-modellering

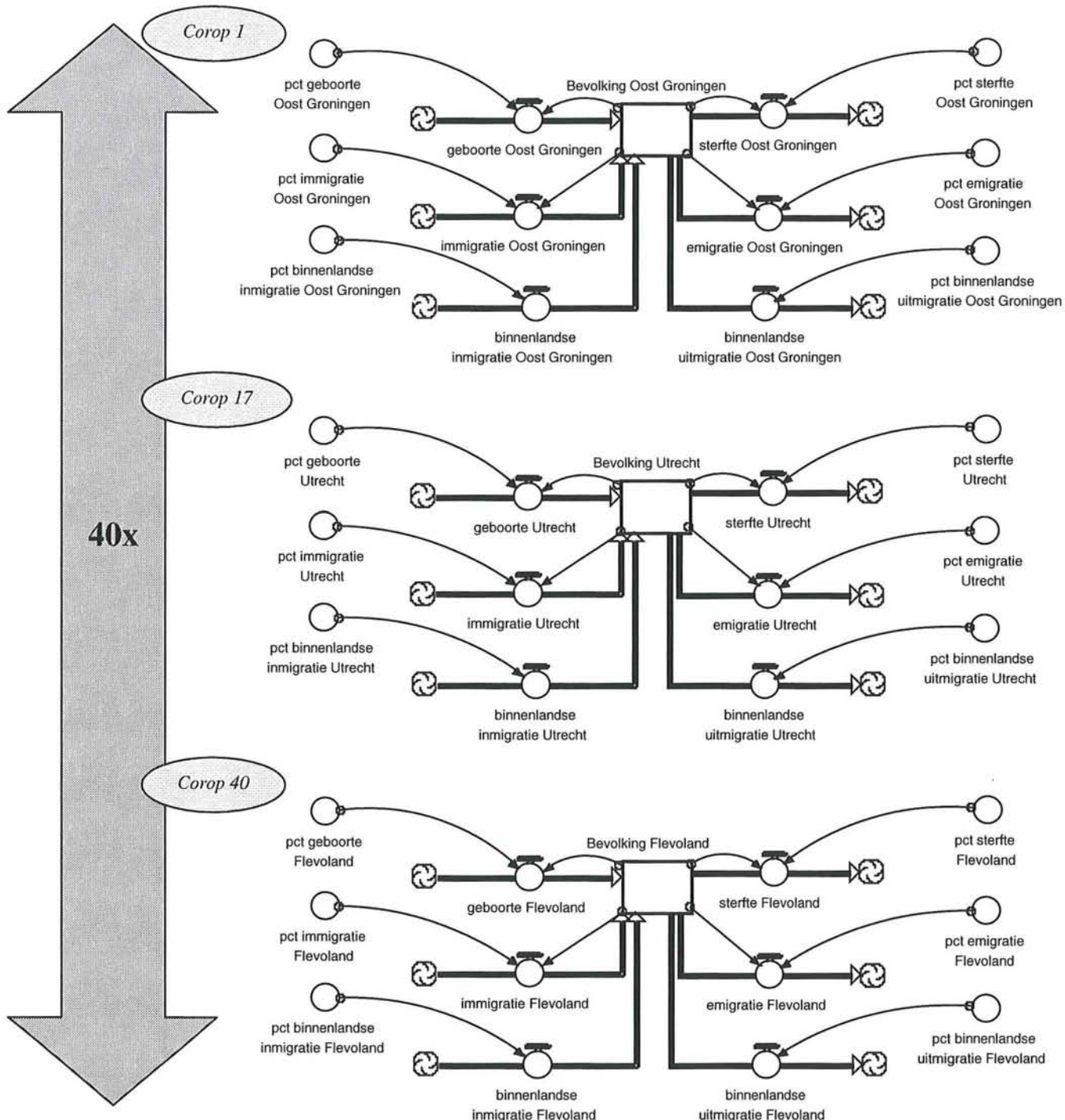
Het bijbehorende Stella-model van het ontwikkelingsproces van de regionale bevolking (hier: regio Utrecht) is weergegeven in figuur-Bijlage 4.4.



Figuur-Bijlage 4.4 Modellering regionale bevolkingsontwikkeling (Utrecht)

**Bijlage 4.1.4 De enkelvoudige array**

De startwaarden en de parameters (de percentages) kunnen voor elke regio anders zijn. De *kwantitatieve* structuur van elk regionaal model is dus niet hetzelfde. De *kwalitatieve* structuur van het regionale bevolkingsmodel is echter wél voor elke regio hetzelfde. In geval veertig regio's moeten worden beschouwd zou men kunnen overwegen om één regionale modelstructuur te bouwen en dit veertig keer te kopiëren. Indien voor elke regio uniek alle namen worden aangepast, levert dit the resultaat op zoals in figuur-Bijlage 4.5 is geïllustreerd.



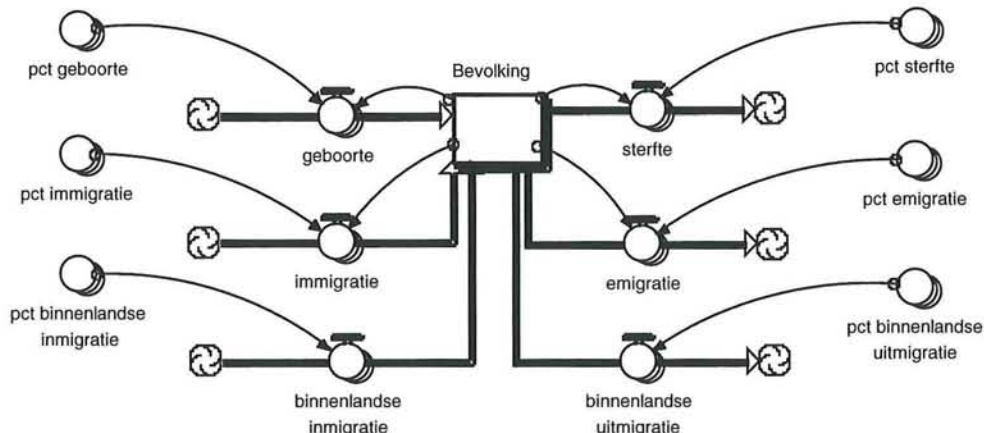
Figuur-Bijlage 4.5 Veertig losse modelstructuren

Aan deze aanpak kleven enkele grote bezwaren:

- a) Het kopiëren van modelstructuur kost veel tijd;
- b) De causale structuur is niet flexibel, aangezien elke structuurverandering 40 keer moet worden doorgevoerd. Dit kost nog veel meer tijd. Het ontwikkelingstraject –welke iteratief van karakter is en veel terugkoppelingen kent– van een model wordt daardoor onmogelijk;
- c) Het totale model wordt erg onoverzichtelijk indien stromen tussen regio's in beeld worden gebracht. De modelstructuur kan hierdoor niet meer onder controle gehouden worden;
- d) De bestandsgrootte van de modelfile wordt erg groot, waardoor Stella langzamer simuleert en in het slechtste geval helemaal niets meer doet.

Array's zijn ervoor bedoeld om **repeterende** modelstructuren op een handige wijze weer te geven. Hierbij kan gedacht worden aan de werking van een *kaartenbak*. In plaats voor elk document een aparte kast in je kamer te hebben, kun je veel handiger één kast met verschillende mappen maken. Dit neemt veel minder ruimte in beslag en is veel overzichtelijker. Het onderscheid in soortgelijke mappen (kwaliteit), elk met een mogelijk andere inhoud (kwantiteit), is noodzakelijk om het systeem (in dit geval de kamer ter opberging van documenten, zoals een bibliotheek) te laten functioneren.

Ook Stella kent array's. In plaats van 40 losse deelmodellen met dezelfde modelstructuur, kan één modelstructuur 40 keer in een *enkelvoudige* array worden gezet. De modellering van figuur-Bijlage 4.5 kan letterlijk vervangen worden door de modellering van figuur-Bijlage 4.6.



Figuur-Bijlage 4.6 Veertig modellen in enkelvoudige array

In figuur-Bijlage 4.6 is gebruik gemaakt van een array. Stella geeft de array weer als een soort kaartenbak. Het multiregionale bevolkingsmodel van figuur-Bijlage 4.6 kan in vier stappen gemaakt worden:

1. Bouw van 1 regionaal bevolkingsmodel;
2. Aanmaken van een array met veertig elementen;
3. Het toepassen van de array-functie uit fase 2 op het regionaal bevolkingsmodel uit fase 1;
4. Het kwantificeren van de regionale bevolkingsontwikkelingen.

Na het maken van een regionaal bevolkingsmodel dient in Stella een array aangemaakt te worden. Dit kan door in Stella op de menubalk <Model> aan te klikken, en vervolgens de <Array-Editor> te openen. Er verschijnt dan een venster met een *Dimension Name* en een *Element Name*. De *Dimension Name* is de naam van de array, de *Element Name* is de naam van een element in die array.

Stel dat alle regio's worden geïndiceerd met  $i$ . De *Dimension Name* wordt  $i$  genoemd. Dan geeft  $i = 1$  de regio *Oost-Groningen* aan en impliceert  $i = 40$  de regio *Flevoland* ( $i = 1...40$ ), conform de huidige Corop-indeling. Er moeten hiervoor 40 *Element Names* worden aangemaakt (40 keer <insert>). Elk regio heeft dan zijn eigen nummer. De regionamen kunnen ook letterlijk worden ingevoerd.

Vervolgens kan voor elke entiteit van het bevolkingsmodel (welke maar één keer fysiek in Stella is gezet) de optie <Array> worden ingeschakeld. Indien een voorraad wordt gearrayed, dan krijgen de flows automatisch dezelfde array-structuur. Indien alle waarden in de array dezelfde waarde hebben, met andere woorden dat element heeft voor elke regio dezelfde waarde, dan kan die waarde ingevoerd worden in combinatie met de afgevinkte optie <Apply To All>. Indien de waarde van een element voor elke regio uniek is moet deze worden ingevoerd door de optie <Apply To All> uit te schakelen, en vervolgens alle bladen van de array af te lopen en de waarde telkens in te vullen. Op dit moment wordt duidelijk dat Stella de array definieert als een soort vector: elke rij correspondeert naar de bijbehorende *Element Name*.

**Voor elke rij  $i$  rekent Stella dezelfde modelstructuur door met de bijbehorende variabelen en toegekende waarden voor  $i$ .**

Het is van belang om te beseffen dat de *modelstructuur* van figuur-Bijlage 4.5 *exact hetzelfde* is als het model van figuur-Bijlage 4.6. Stella rekent precies hetzelfde uit. De gebruiker ondervindt echter een paar essentiële voordelen van de array-aanpak:

- Het bouwen van een multiregionale modelstructuur kost weinig tijd;
- de causale structuur kan voor veertig regio's in één keer aangepast worden, waardoor modelontwikkeling mogelijk wordt;
- Het gearrayde model is *visueel* kleiner van omvang, waardoor de modellering overzichtelijker en onder controle te houden is;
- Het gearrayde model is *fysiek* kleiner van omvang, waardoor het model sneller rekent en minder bestandsgrootte vergt. Bij omvangrijke modellen kan dit het verschil maken tussen het nog-net-wel en niet-meer functioneren van Stella.

#### **Bijlage 4.1.5 De dubbele array**

Naast het onderscheid naar 40 regio's kan de bevolking bijvoorbeeld worden gesegmenteerd naar inkomensklassen. Indien het bevolkingsmodel niet alleen per regio dezelfde structuur heeft, maar ook voor elke inkomensklasse dezelfde structuur heeft ontstaat wederom de wens om te arrayen, waardoor repeterende elementen worden voorkomen.

Op dit moment kan de **dubbele array** toegepast worden. Naast de eerste *Dimension Name*  $i$ , kan een tweede *Dimension Name* worden aangemaakt (met <Array-Editor>): *inkomensklasse*. Elke *Element Name* in deze tweede array komt overeen met een inkomensklasse.

Vervolgens kan binnen elke entiteit van het model gekozen worden voor de optie <2d>. Naast een rij maakt Stella dan ook een kolom aan. De gebruiker kan zelf instellen welke waarde op de rij moet staan en welke waarde op de kolommen worden weergegeven. Op papier kan de dubbele array worden geïnterpreteerd als een *matrix*.

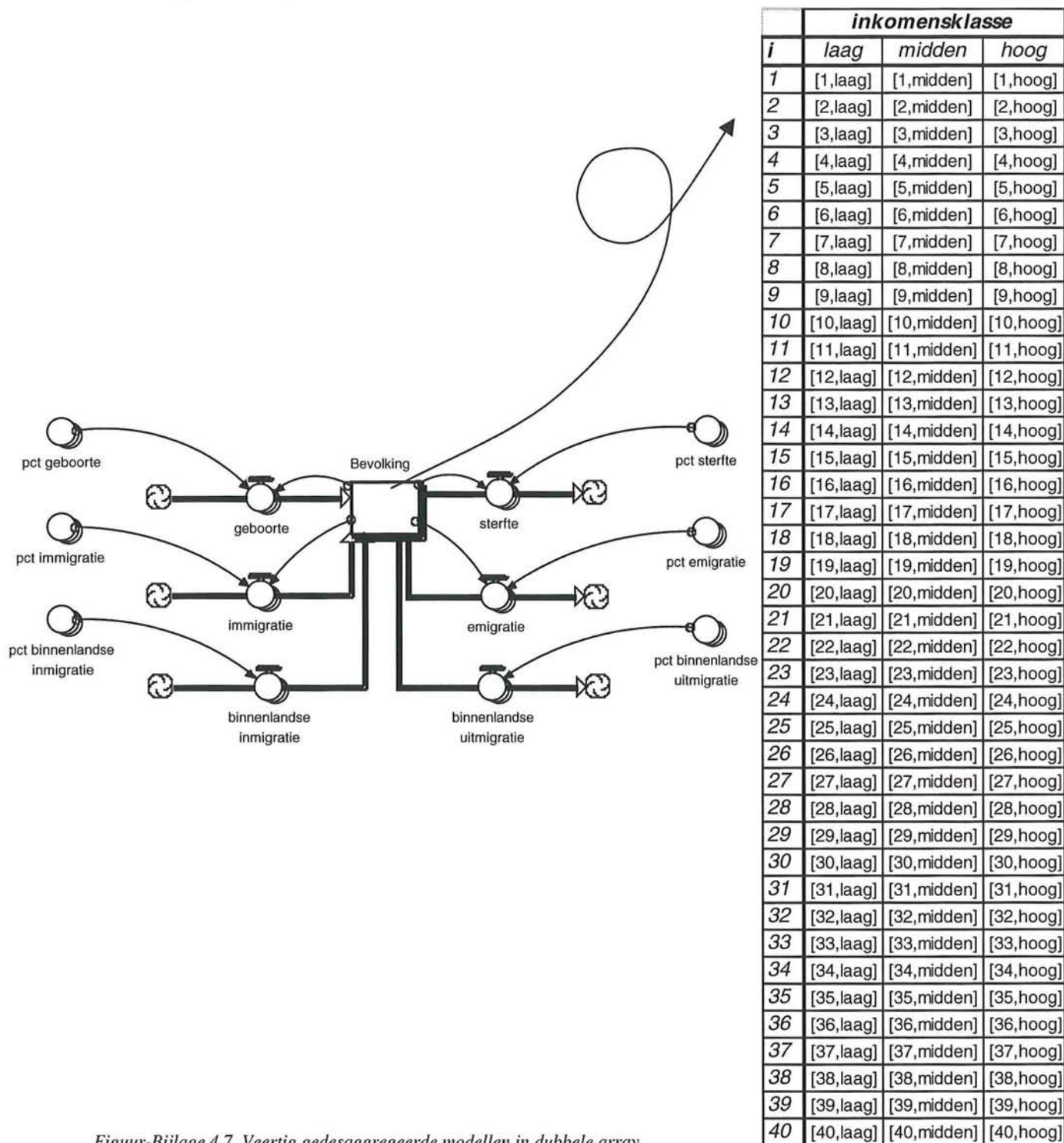
**De gebruiker dient zelf te beseffen wat de interpretatie is van elke waarde in de dubbele array.**

In het voorbeeld wordt de voorraad bevolking een twee-dimensionaal karakter gegeven. Voor 40 regio's en voor 3 inkomensklassen is de modelstructuur hetzelfde. Er zijn veertig regio's ontstaan doordat de eerste *Dimension Name* is gedefinieerd als  $i$ , met  $i = 1...40$ . De tweede *Dimension Name* is gedefinieerd als *inkomensklasse*, met *inkomensklasse* = laag, midden, hoog. Via de optie



<2d> is als rij (de linker-optie naast 'to editor') *i* ingesteld, en als kolom (de rechter optie naast 'to editor') *inkomensklasse* ingesteld.

Op de *Map/Model-level* van Stella lijkt het model *op het oog* niet veranderd. Echter, wordt een variabele aangeklikt dan kan men zien dat er een dubbele array is ingevoerd. Voor elke combinatie van regio en inkomensklasse kan een unieke waarde worden ingevoerd (indien <Apply To All> is uitgeschakeld). De interpretatie van elke rij en kolom wordt duidelijk in figuur-Bijlage 4.7. In deze figuur is de feitelijke inhoud van de voorraad bevolking weergegeven. Alles is geïndiceerd. Bevolking[1,laag] correspondeert met het aantal mensen in regio 1 in de bevolkingsklasse laag. Bevolking [17,hoog] correspondeert met het aantal mensen in regio 17 in de bevolkingsklasse hoog.



Figuur-Bijlage 4.7 Veertig gedesaggregeerde modellen in dubbele array

## Bijlage 4.2 Creatieve toepassingen van array's in Stella

De aanwezigheid van dubbele array's maakt het mogelijk om binnen Stella matrixberekeningen uit te voeren. Deze stelling impliceert dat indien bestaande methoden en technieken geënt zijn op matrixberekeningen, deze methoden en technieken op eenvoudige wijze in een dynamische omgeving kunnen worden geïmplementeerd. Dit betekent dat onder andere *zwaartekrachtmodellen* en *Input-Output modellen* op eenvoudige wijze **dynamisch** kunnen worden toegepast.

Het instrument 'dubbele array' in combinatie met enige creativiteit maakt het mogelijk om in Stella zeer complexe systemen te bouwen. Er kan bijvoorbeeld worden gekozen om de ruimtelijke component van processen ermee dynamisch in beeld te brengen. De modellering blijft dankzij de array's overzichtelijk en praktisch hanteerbaar.

In deze bijlage wordt uitgelegd hoe enkele interessante en complexe toepassingen in Stella kunnen worden gezet:

- Hansen's Potential Model (paragraaf 4.2.1);
- Een double-constrained zwaartekrachtmodel (paragraaf 4.2.2);
- Overige modeltechnieken (paragraaf 4.2.3).

De toepassing van de array-techniek wordt aan de hand van een bestaand voorbeeld (Hansen's Potential Model) stap voor stap geïllustreerd. Een voorbeeld van een double-constrained zwaartekrachtmodel is beknopt gegeven. Ten slotte wordt een overzicht gegeven van enkele andere toepassingen die simpel en snel dynamisch kunnen worden toegepast.

### Bijlage 4.2.1 Hansen's Potential Model in Stella

#### *Definiëring van de causaliteit*

Hansen's model is een locatiemodel om de verdeling van de bevolking over regio's te voorspellen, gebaseerd op de veronderstelling dat werkgelegenheid en de beschikbaarheid van woonruimte de belangrijkste bepalende factoren hierin zijn.

Volgens Hansen zou de relatie tussen de verdeling van de bevolking en de werkgelegenheid kunnen worden uitgedrukt in een bereikbaarheidsindex ( $A_i$ ), die voor elke regio de bereikbaarheid van de werkgelegenheid weergeeft.

Vanuit elke regio wordt bekeken wat de *aantrekkingskracht* of *massa* (hier het aantal banen) van elke andere regio is. Indien ergens veel banen zijn is het aannemelijk te veronderstellen dat er veel mensen wonen. Hansen veronderstelt dat ook de bereikbaarheid van de massa van belang is. Als ergens veel banen zijn gecreëerd, maar indien de regio slecht bereikbaar is dan is het onwaarschijnlijk dat veel mensen hierop af zijn gekomen. Hansen veronderstelt dat de bereikbaarheid omgekeerd evenredig is met de afstand  $d$  met een bijbehorende, universele weerstandsfactor  $b$ .

De bereikbaarheidsindex van regio  $i$  in relatie tot regio  $j$  is dus gelijk aan de massa van regio  $j$  gedeeld door de afstand  $d$  met een universele weerstandsfactor  $b$ . In wiskundige notatie is dit:

$$A_{ij} = \frac{E_j}{d_{ij}^b} \quad (0.38)$$

Waarin:

$A_{ij}$  = bereikbaarheidsindex van regio  $i$  in relatie tot regio  $j$ ;

- $E_j$  = totale werkgelegenheid in regio  $j$ ;  
 $d_{ij}$  = afstand tussen regio  $i$  en  $j$ ;  
 $b$  = weerstandsfactor behorende bij  $d_{ij}$ .

De som van alle relatieve bereikbaarheidsindices is voor elke regio een maat voor de bereikbaarheid van banen vanuit die regio:

$$A_i = \sum_j \frac{E_j}{d_{ij}^b} \quad (0.39)$$

Waarin:

- $A_i$  = bereikbaarheidsindex van regio  $i$ ;  
 $E_j$  = totale werkgelegenheid in regio  $j$ ;  
 $d_{ij}$  = afstand tussen regio  $i$  en  $j$ ;  
 $b$  = weerstandsfactor behorende bij  $d_{ij}$ .

Het aantal mensen dat door een regio wordt aangetrokken wordt naast de bereikbaarheid van banen ook bepaald door de beschikbaarheid van woonruimte. Hansen noemt dit de wooncapaciteit ( $H_i$ ) van een gebied. Een maat voor  $H_i$  is het aantal woningen in een regio. Door vermenigvuldiging van de bereikbaarheidsindex met de wooncapaciteit kan een index voor de ontwikkelingspotentie ( $D_i$ ) van een regio worden verkregen.

$$D_i = A_i H_i \quad (0.40)$$

Waarin:

- $D_i$  = ontwikkelingspotentie van regio  $i$ ;  
 $A_i$  = bereikbaarheidsindex van regio  $i$ ;  
 $H_i$  = woningvoorraad in regio  $i$ .

De veronderstelling is dat in een regio met een relatief hoge waarde voor  $D_i$  veel mensen zullen wonen. Vanuit deze regio zijn banen relatief goed bereikbaar en zijn relatief veel potentiële woningen aanwezig. De *relatieve* ontwikkelingspotentie van een regio wordt bepaald door het aandeel van de ontwikkelingspotentie van de regio in het totaal aan ontwikkelingspotenties.

De totale bevolking van alle regio's samen, die bekend wordt verondersteld, wordt gealloceerd op basis van de relatieve ontwikkelingspotentie van elke regio. De verdeling (allocatie) van de bevolking over regio's is zodoende gebaseerd op de aantrekkelijkheid van elke regio t.o.v. de andere (concurrerende) regio's.

$$G_i = G_r \frac{D_i}{\sum_i D_i} = G_r \frac{(A_i H_i)}{\left( \sum_i A_i H_i \right)} \quad (0.41)$$

Waarin:

- $G_i$  = bevolkingsgroei van regio  $i$ ;  
 $G_r$  = totale bevolkingsgroei van Nederland;  
 $D_i$  = ontwikkelingspotentie van regio  $i$ ;  
 $A_i$  = bereikbaarheidsindex van regio  $i$ ;  
 $H_i$  = woningvoorraad in regio  $i$ .

### ***Het gehanteerde voorbeeld***

Het gekozen voorbeeld is oorspronkelijk overgenomen uit *Models in planning* (Lee, C. (1973), *An introduction to the use of quantitative models in planning*, Newcastle upon Tyne). In deze uitgave wordt een voorbeeld van toepassing van Hansen's Potential Model uitgewerkt. Het voorbeeld is tevens terug te vinden in het collegedictaat behorende bij het vak *Ruimtelijke inrichting* (CTip3070, uitgave oktober 1998).

In dit voorbeeld wordt Hansen's Potential Model gebruikt om de huidige bevolkingsverdeling te reproduceren. Hiervoor is de huidige werkgelegenheid als input genomen om na te gaan of het model de huidige verdeling van de bevolking juist kan voorspellen. Er wordt onderscheid gemaakt in drie zones, waarvan de gegevens in tabel bijlage 4.1 en tabel bijlage 4.2 zijn weergegeven.

Tabel-Bijlage 4.1 Voorbeeldgegevens Hansen-model

zone	Totale werkgelegenheid	Totale bevolking	Wooncapaciteit (ha)
1	4000	19000	100
2	8000	35000	125
3	32000	41000	100
Totaal	44000	95000	325

Tabel-Bijlage 4.2 Afstands-/reistijdenmatrix

Naar j			
	$j = 1$	$j = 2$	$j = 3$
Van i			
$i = 1$	2	8	6
$i = 2$	8	3	4
$i = 3$	6	4	3

### Toepassing Hansen's Potential Model

Allereerst worden alle bereikbaarheidsindices  $A_{ij}$  berekend, waarbij de parameter  $b$  is gelijkgesteld aan 2. Sommatie van  $A_{ij}$  voor elke regio  $i$  levert de bereikbaarheidsindex  $A_i$  voor regio  $i$  op. De berekening is weergegeven in tabel-Bijlage 4.3.

Tabel-Bijlage 4.3 Berekening  $A_{ij}$  en  $A_i$ 

zone	$j = 1$	$j = 2$	$j = 3$	$\sum_j A_{ij}$
$i = 1$	$\frac{4000}{2^2} = 1000$	$\frac{8000}{8^2} = 125$	$\frac{32000}{6^2} = 888,89$	2014
$i = 2$	$\frac{4000}{8^2} = 62,5$	$\frac{8000}{3^2} = 888,89$	$\frac{32000}{4^2} = 2000$	2951
$i = 3$	$\frac{4000}{6^2} = 111,1$	$\frac{8000}{4^2} = 500$	$\frac{32000}{3^2} = 3555,56$	4167

Vervolgens kan de ontwikkelingspotentie  $D_i$  van elke zone worden bepaald (tabel-Bijlage 4.4).

Tabel-Bijlage 4.4 Berekening  $D_i$ 

zone	$A_i$	$H_i$	$D_i$
1	2014	100	201400
2	2951	125	368924
3	4167	100	416700
Totaal			987024

De verdeling van de totale bevolking (95000) geschiedt op basis van de relatieve ontwikkelingspotentie  $D_i$ . De eindresultaten zijn weergegeven in tabel-Bijlage 4.5.

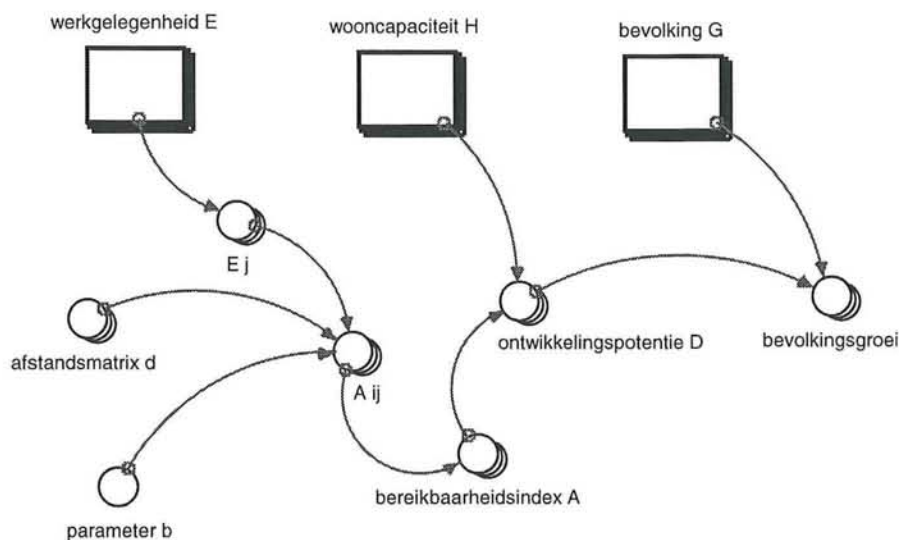
Tabel-Bijlage 4.5 Bepaling bevolkingsverdeling

zone	$D_i$	$\frac{D_i}{\sum_i D_i}$	$G_i = G_t \left( \frac{D_i}{\sum_i D_i} \right)$
1	201400	0,204	$0,204 * 95000 = 19384$
2	368924	0,374	$0,374 * 95000 = 35510$
3	416700	0,422	$0,422 * 95000 = 40106$
Totaal	987024	1	95000

### Stella-modellering

Het Hansen-model rekent met matrices. Een matrix kan in Stella geassocieerd worden met een dubbele array. Het is dus mogelijk om het Hansen-model in Stella te zetten. Daarmee kan het model dynamisch toegepast worden.

Voor elke zone zijn drie entiteiten bekend verondersteld: de *werkgelegenheid* ( $E$ ), de *bevolking* ( $G$ ) en de *wooncapaciteit* ( $H$ ). Hiervoor worden in Stella drie entiteiten aangemaakt, in dit geval in de vorm van drie voorraadgrootheden. Een totaal overzicht van alle entiteiten die in Stella nodig zijn voor de weergave van het Hansen model is weergegeven in figuur-Bijlage 4.8.



Figuur-Bijlage 4.8 Overzicht Stella-model van Hansen's Potential Model

In plaats van de drie voorraden 3 keer (voor elke zone) te kopiëren worden de drie zones ingevoerd met behulp van een array. In de array-editor wordt een dimensie  $i$  aangemaakt, met drie elementen (de zones 1, 2 en 3). In elke voorraad uit figuur-Bijlage 4.8 wordt de array-optie ingeschakeld voor de bestaande array  $i$ , met <Apply-To-All> uitgeschakeld.

Na de invoering van de drie voorraden kan voor elke regio uniek de waarde van elke voorraad worden ingevoerd op de bijbehorende rij in de voorraad:

$E[1] = 4000$	$G[1] = 19000$	$H[1] = 100$
$E[2] = 8000$	$G[2] = 35000$	$H[2] = 125$
$E[3] = 32000$	$G[3] = 41000$	$H[3] = 100$

Vervolgens dient de afstandsmatrix te worden ingevoerd. Een matrix kan in Stella worden ingevoerd door middel van een dubbele array. In het gehanteerde voorbeeld bestaat de afstandsmatrix uit een rij  $i$  en een kolom  $j$  (zie tabel-Bijlage 4.2). De betekenis van de rij voor  $i$  is

reeds in Stella ingevoerd. Er dient nu een tweede dimensie te worden ingevoerd:  $j$ . De dimensie  $j$  heeft ook drie elementen (1, 2 en 3).

Er wordt in Stella nu een entiteit  $d$  ingevoerd. Hier wordt de dubbele array in ingeschakeld, met op de rij  $i$  (links) en op de kolom  $j$  (rechts) en met <Apply-To-All> uitgeschakeld. Vervolgens kunnen de onderlinge afstanden uniek worden ingevoerd:

$$\begin{array}{lll} d[1,1] = 2 & d[1,2] = 8 & d[1,3] = 6 \\ d[2,1] = 8 & d[2,2] = 3 & d[2,3] = 4 \\ d[3,1] = 6 & d[3,2] = 4 & d[3,3] = 3 \end{array}$$

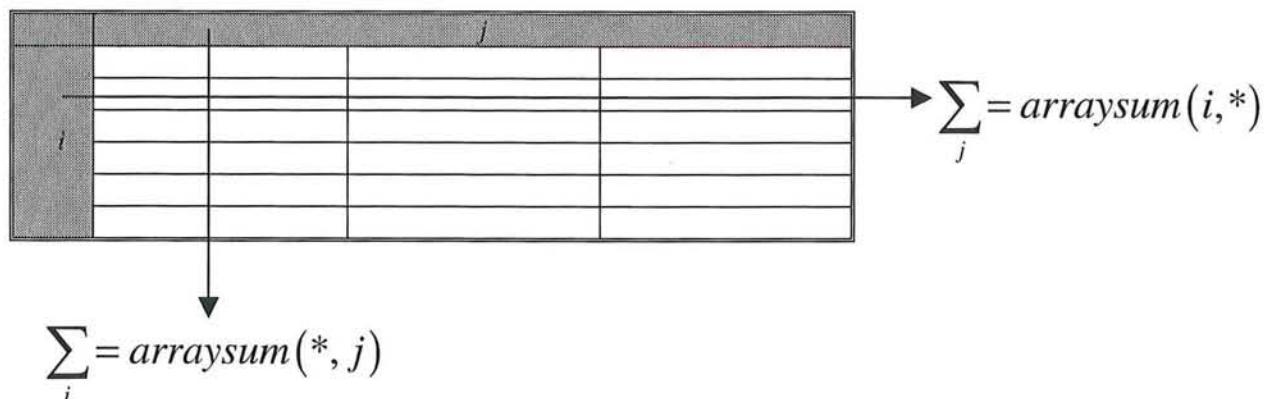
Als laatste moet parameter  $b$  worden ingevoerd. Dit is een 'normale' variabele met de waarde 2.

Alle bekende gegevens zijn nu ingevoerd. Nu dient Stella de berekening uit te voeren. Als eerste wordt  $A_{ij}$  bepaald. Hiervoor wordt de entiteit  $A_{ij}$  in Stella aangemaakt. Zoals uit tabel-Bijlage 4.3 blijkt, vindt de berekening van  $A_{ij}$  in matrixvorm plaats met in elke cel dezelfde wiskundige formulering (formule 0.38). In Stella dient dezelfde gedachtegang te worden gehanteerd. De entiteit  $A_{ij}$  wordt in array tweedimensionaal gemaakt (net als  $d$ ), maar nu met <Apply-To-All> ingeschakeld. De wiskundige formulering is namelijk voor elke cel in  $A_{ij}$  hetzelfde.

In deze wiskundige formulering is echter  $E_j$  gewenst. Deze is nog niet in het model aanwezig. De ingevoerde waarden voor  $E$  zijn namelijk in de vorm (array)  $i$ . Zoals uit tabel-Bijlage 4.3 blijkt wordt dezelfde waarde voor  $E$  gebruikt in het stramien van  $j$  en niet in dat van  $i$ . Daarom dient  $E_j$  aangemaakt te worden: een hulpvariabele met de dimensie  $j$ . Nu dienen de rij-waarden uit de oorspronkelijke entiteit  $E_i$  op de kolommen te komen staan in de nieuwe entiteit  $E_j$ . Dit moet uniek **handmatig** worden ingevoerd. Nu kan de algemene formule 0.38 letterlijk in de variabele  $A_{ij}$  ingevoerd worden.

Vervolgens wordt formule 0.39 bepaald. Uit  $A_{ij}$  dient de waarde voor  $A_i$  afgeleid te worden. Zoals tabel-Bijlage 4.3 duidelijk maakt, dienen voor de rijwaarde in  $A_i$  van de overeenkomende rijwaarde in  $A_{ij}$  de kolomwaarden van  $A_{ij}$  opgesomd te worden. Hiervoor bestaat een Stella een speciale functie: *arraysum*. Deze sommage wordt in Stella aangeduid met een asterisk \*.  $A_i$  wordt bepaald door  $i$  van  $A_{ij}$  te behouden, en vervolgens de  $j$ -waarden van  $A_{ij}$  met een \* op te sommen. Dit is de algemene formule 0.39.

Het denkpatroon bij het maken van sommages in Stella is weergegeven in figuur-Bijlage 4.9. Een 'arraysum' van (i,\*) betekend dat van een bepaalde matrix voor elke rij  $i$  alle waarden van de kolommen  $j$  worden opgeteld. Bij een 'arraysum' (\*,j) wordt van elke kolom  $j$  in een bepaalde matrix de waarden van alle rijen  $i$  opgeteld.



Figuur-Bijlage 4.9 Sommeren in de array-structuur

Vervolgens moet om de ontwikkelingspotentie  $D_i$  uit te rekenen (formule 0.40) een nieuwe variabele aangemaakt worden met de arraystructuur  $i$ . De ontwikkelingspotentie  $D_i$  kan in één keer worden bepaald met de algemene formule 0.40.

De laatste stap is formule 0.41. Hiervoor wordt een nieuwe variabele *bevolkingsgroei* aangemaakt. Hierin kan de algemene formule 0.41 in ingevoerd worden. Voor de formulering wordt verwezen naar onderstaande Stella-listing.

Opgemerkt wordt dat de berekeningen in het Hansen-model in minder variabelen dan hierboven beschreven kan worden. Voor de overzichtelijkheid zijn echter de tussenberekeningen getoond. Op deze manier wordt getracht de communicatieve waarde van System Dynamics enigszins te behouden.

De invoervariabelen voor het model kunnen vervolgens in de tijd exogeen of endogeen in modellering worden gebracht, waardoor het Hansen Potential Model *dynamisch* kan worden gebruikt, in tegenstelling tot bijvoorbeeld Excel waar alleen statische berekeningen kunnen worden gemaakt.

### ***Stella-listing voorbeeld Hansen-model***

Hieronder volgt de Stella-listing van het toegelichte voorbeeld van het Hansen Potential Model.

```

bevolking_G[1](t) = bevolking_G[1](t - dt)
INIT bevolking_G[1] = 19000
bevolking_G[2](t) = bevolking_G[2](t - dt)
INIT bevolking_G[2] = 35000
bevolking_G[3](t) = bevolking_G[3](t - dt)
INIT bevolking_G[3] = 41000
werkgelegenheid_E[1](t) = werkgelegenheid_E[1](t - dt)
INIT werkgelegenheid_E[1] = 4000
werkgelegenheid_E[2](t) = werkgelegenheid_E[2](t - dt)
INIT werkgelegenheid_E[2] = 8000
werkgelegenheid_E[3](t) = werkgelegenheid_E[3](t - dt)
INIT werkgelegenheid_E[3] = 32000
wooncapaciteit_H[1](t) = wooncapaciteit_H[1](t - dt)
INIT wooncapaciteit_H[1] = 100
wooncapaciteit_H[2](t) = wooncapaciteit_H[2](t - dt)
INIT wooncapaciteit_H[2] = 125
wooncapaciteit_H[3](t) = wooncapaciteit_H[3](t - dt)
INIT wooncapaciteit_H[3] = 100
afstandsmatrix_d[1,1] = 2
afstandsmatrix_d[1,2] = 8
afstandsmatrix_d[1,3] = 6
afstandsmatrix_d[2,1] = 8
afstandsmatrix_d[2,2] = 3
afstandsmatrix_d[2,3] = 4
afstandsmatrix_d[3,1] = 6
afstandsmatrix_d[3,2] = 4
afstandsmatrix_d[3,3] = 3
A_ij[i,j] = E_j[j]/afstandsmatrix_d[i,j]^parameter_b
bereikbaarheidsindex_A[i] = ARRAYSUM(A_ij[i,*])
bevolkingsgroei[i] =
ARRAYSUM(bevolking_G[*])*ontwikkelingspotentie_D[i]/ARRAYSUM(ontwikkelingspotenti
e_D[*])
E_j[1] = werkgelegenheid_E[1]
E_j[2] = werkgelegenheid_E[2]

```

$E_j[3] = \text{werkgelegenheid}_E[3]$

$\text{ontwikkelingspotentie}_D[i] = \text{bereikbaarheidsindex}_A[i] * \text{wooncapaciteit}_H[i]$

$\text{parameter}_b = 2$



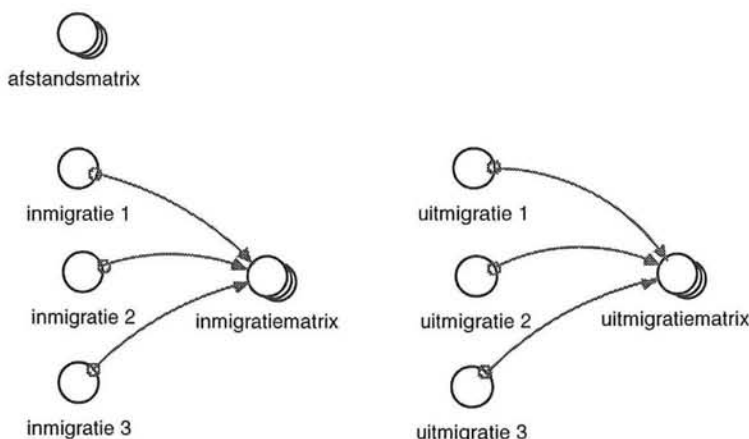
**Bijlage 4.2.2 Een double-constrained model in Stella**

In Bijlage 3.2 is de wiskundige achtergrond van een double-constrained model toegelicht. Voor een uitgebreidere theoretische toelichting wordt verwezen naar de volgende literatuurbron: Bovy, P.H.L., Zijpp, N.J. van der, (2001), *Course trail transportation modelling*, The Netherlands Research School for transport, infrastructure and logistics (TRAIL), Delft, Nederland.

Hieronder wordt van een double-constrained model een voorbeeld gegeven. Hiertoe is (1) een overzicht en (2) de listing van het Stella-model gegeven. Op basis hiervan kan de benadering gereproduceerd worden.

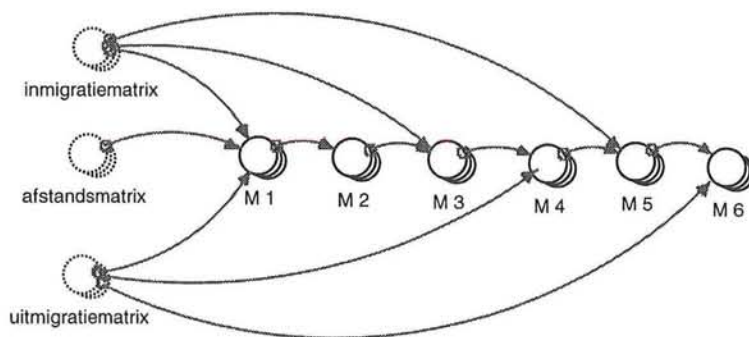
**Double-constrained voorbeeld in Stella**

In figuur-Bijlage 4.10 is een Stella-overzicht van de invoergegevens van een double-constrained zwaartekrachtmodel gegeven. De specificering hiervan volgt in de listing aan het einde van deze subparagraaf..



*Figuur-Bijlage 4.10 Invoergegevens voor een voorbeeld van een double-constrained zwaartekrachtmodel*

De iteratieve oplossingsprocedure die hiermee gemaakt moet worden bestaat uit matrixberekeningen ( $M$ ) en kan dus in Stella worden geïmplementeerd. Het Stella-overzicht van de iteratieve oplossingsprocedure is in figuur-Bijlage 4.11 weergegeven.



*Figuur-Bijlage 4.11 Iteratieve oplossingsprocedure van een double-constrained zwaartekrachtmodel*

**Stella-listing double-constrained voorbeeld**

```

afstandsmatrix[1,1] = 10
afstandsmatrix[1,2] = 100
afstandsmatrix[1,3] = 50
afstandsmatrix[2,1] = 100
afstandsmatrix[2,2] = 10
afstandsmatrix[2,3] = 80
afstandsmatrix[3,1] = 50
afstandsmatrix[3,2] = 80
afstandsmatrix[3,3] = 10
inmigratiematrix[1,1] = inmigratie_1+0*(inmigratie_2+inmigratie_3)
inmigratiematrix[1,2] = inmigratie_1+0*(inmigratie_2+inmigratie_3)
inmigratiematrix[1,3] = inmigratie_1+0*(inmigratie_2+inmigratie_3)
inmigratiematrix[2,1] = inmigratie_2+0*(inmigratie_1+inmigratie_3)
inmigratiematrix[2,2] = inmigratie_2+0*(inmigratie_1+inmigratie_3)
inmigratiematrix[2,3] = inmigratie_2+0*(inmigratie_1+inmigratie_3)
inmigratiematrix[3,1] = inmigratie_3+0*(inmigratie_2+inmigratie_1)
inmigratiematrix[3,2] = inmigratie_3+0*(inmigratie_2+inmigratie_1)
inmigratiematrix[3,3] = inmigratie_3+0*(inmigratie_2+inmigratie_1)
inmigratie_1 = 100000
inmigratie_2 = 300000
inmigratie_3 = 200000
uitmigratiematrix[1,1] = uitmigratie_1+0*(uitmigratie_2+uitmigratie_3)
uitmigratiematrix[1,2] = uitmigratie_2+0*(uitmigratie_1+uitmigratie_3)
uitmigratiematrix[1,3] = uitmigratie_3+0*(uitmigratie_2+uitmigratie_1)
uitmigratiematrix[2,1] = uitmigratie_1+0*(uitmigratie_2+uitmigratie_3)
uitmigratiematrix[2,2] = uitmigratie_2+0*(uitmigratie_1+uitmigratie_3)
uitmigratiematrix[2,3] = uitmigratie_3+0*(uitmigratie_2+uitmigratie_1)
uitmigratiematrix[3,1] = uitmigratie_1+0*(uitmigratie_2+uitmigratie_3)
uitmigratiematrix[3,2] = uitmigratie_2+0*(uitmigratie_1+uitmigratie_3)
uitmigratiematrix[3,3] = uitmigratie_3+0*(uitmigratie_2+uitmigratie_1)
uitmigratie_1 = 400000
uitmigratie_2 = 100000
uitmigratie_3 = 100000
M_1[van,naar] =
(inmigratiematrix[van,naar]*uitmigratiematrix[van,naar])/(afstandsmatrix[van,naar]^2)
M_2[van,naar] = (M_1[van,naar]*600)/ARRAYSUM(M_1[*,*])
M_3[van,naar] = (M_2[van,naar]/ARRAYSUM(M_2[van,*]))*inmigratiematrix[van,naar]
M_4[van,naar] = (M_3[van,naar]/ARRAYSUM(M_3[*,naar]))*uitmigratiematrix[van,naar]
M_5[van,naar] = (M_4[van,naar]/ARRAYSUM(M_4[van,*]))*inmigratiematrix[van,naar]
M_6[van,naar] = (M_5[van,naar]/ARRAYSUM(M_5[*,naar]))*uitmigratiematrix[van,naar]

```

**Bijlage 4.2.3 Overige modeltechnieken in Stella**

Ten aanzien van overige toepassingen moet bedacht worden dat:

1. Met tweedimensionale array's matrixberekeningen kunnen worden uitgevoerd. Dit impliceert onder meer dat indien er veel onderlinge stromen tussen entiteiten bestaan, zoals tussen ruimtelijke gebieden of tussen bedrijfssectoren, deze vereenvoudigd kunnen worden weergegeven;
2. In Stella dynamische analyses gemaakt kunnen worden.

Enkele andere interessante bestaande modeltechnieken die hierdoor gemakkelijk en snel in Stella –en daarmee in een dynamische omgeving- kunnen worden toegepast zijn onder andere:

- Input-Output modellen;
- Cohort-survival modellen;
- Shift-Share analyses;
- Entropie;
- Overige zwaartekrachtmodellen;
- Informatievelden;
- ‘Distance decay’;
- Logit-formuleringen.

Bovenstaande modeltechnieken worden in bijlage 3 kort toegelicht. Van de toegepaste entiteiten kunnen invoerwaarden die veranderen in de tijd exogeen in een trend worden gezet. Het meest interessante resultaat ontstaat als deze parameters endogeen en met wisselwerking in beeld worden gebracht.

Opgemerkt wordt dat andere software-applicaties bepaalde voordelen hebben ten opzichte van Stella. Excel, waar statische modellen in kunnen worden gemaakt, heeft bijvoorbeeld de mogelijkheid om parameters te optimaliseren (‘solven’). Indien een statische optimalisatie in Excel wordt gemaakt, kan daarna deze parameterwaarde in het zelfde model, maar dan in Stella, worden ingevoerd. Indien in Excel de parameter voor verschillende tijdstippen is geoptimaliseerd, kan een trend in de ontwikkeling van deze parameterwaarde in Stella worden ingevoerd waarmee de optimale simulatie wordt nagebootst.



## **BIJLAGEN HOOFDSTUK 5**

### **INVOERGEGEVENS VAN HANSEN'S POTENTIAL MODEL**



Tabel-Bijlage 5.1 Invoer toestandsvariabelen van de Hansen-analyses

Corop	1998		1999			
	banen	bevolking	banen ( $E_j$ )	bevolking ( $E_j$ )	woningen ( $D_i$ )	vestigingen ( $D_i$ )
1	40.600	152.550	41.700	152.651	64.783	6.510
2	16.200	53.323	15.700	52.899	23.151	2.200
3	140.400	352.144	144.100	354.438	155.113	16.295
4	109.500	322.256	113.600	322.682	135.641	15.450
5	28.900	99.736	28.800	100.954	42.270	5.915
6	60.100	196.123	61.600	197.586	81.583	9.860
7	56.000	175.422	60.200	176.515	71.603	7.950
8	47.100	165.582	51.700	166.103	67.887	7.355
9	46.800	123.668	41.400	124.475	50.161	6.375
10	122.900	339.252	129.300	342.548	131.993	17.385
11	50.500	132.789	52.600	134.145	54.167	6.115
12	208.400	591.486	214.400	593.751	233.965	27.310
13	237.600	622.474	246.200	624.776	230.435	30.535
14	128.000	374.807	131.200	375.953	146.259	20.610
15	260.700	680.463	267.700	685.689	280.755	28.215
16	65.800	217.912	69.300	220.360	81.611	13.630
17	503.400	1.088.621	526.100	1.098.722	440.409	54.300
18	105.700	349.428	108.900	351.061	140.259	18.625
19	75.100	227.608	77.500	229.090	92.650	10.895
20	60.000	170.676	60.800	171.588	71.792	7.390
21	74.100	216.381	74.000	216.773	95.723	10.680
22	43.800	149.412	44.800	150.061	62.336	6.155
23	615.800	1.139.976	648.000	1.151.830	543.841	66.645
24	93.300	232.624	93.800	232.755	100.130	13.555
25	134.200	380.941	138.400	382.901	151.667	17.110
26	325.300	709.059	336.400	711.188	330.128	31.120
27	91.700	225.607	96.200	228.516	91.524	12.575
28	105.500	319.599	108.700	320.864	123.110	16.595
29	513.200	1.317.310	524.600	1.326.588	586.028	57.005
30	147.600	406.531	152.500	408.713	164.952	17.930
31	37.300	107.274	37.700	107.058	49.553	6.305
32	83.000	262.675	83.800	263.513	112.372	13.490
33	224.500	581.764	232.900	585.663	238.546	29.165
34	156.800	429.608	161.500	436.124	171.589	21.480
35	240.800	607.958	245.200	611.360	233.909	30.850
36	282.200	699.932	294.400	704.562	283.046	34.135
37	102.700	270.820	105.700	271.717	106.796	13.495
38	76.000	217.623	78.200	218.412	87.428	11.315
39	229.600	649.492	232.700	649.173	275.082	25.640
40	78.900	293.286	83.900	306.468	118.115	13.200
<b>Totaal</b>	<b>6.020.000</b>	<b>15.654.192</b>	<b>6.216.200</b>	<b>15.760.225</b>	<b>6.522.362</b>	<b>761.365</b>

(bron: Statline, CBS, 2001)

Tabel-Bijlage 5.2 Ontwikkeling invoer toestandsvariabelen van de Hansen-analyses

Corop	periode 1998-1999			
	absolute groei bevolking	relatieve groei bevolking	absolute groei werkgelegenheid	relatieve groei werkgelegenheid
1	101	0%	1.100	3%
2	-424	-1%	-500	-3%
3	2.294	1%	3.700	3%
4	426	0%	4.100	4%
5	1.218	1%	-100	0%
6	1.463	1%	1.500	2%
7	1.093	1%	4.200	8%
8	521	0%	4.600	10%
9	807	1%	-5.400	-12%
10	3.296	1%	6.400	5%
11	1.356	1%	2.100	4%
12	2.265	0%	6.000	3%
13	2.302	0%	8.600	4%
14	1.146	0%	3.200	3%
15	5.226	1%	7.000	3%
16	2.448	1%	3.500	5%
17	10.101	1%	22.700	5%
18	1.633	0%	3.200	3%
19	1.482	1%	2.400	3%
20	912	1%	800	1%
21	392	0%	-100	0%
22	649	0%	1.000	2%
23	11.854	1%	32.200	5%
24	131	0%	500	1%
25	1.960	1%	4.200	3%
26	2.129	0%	11.100	3%
27	2.909	1%	4.500	5%
28	1.265	0%	3.200	3%
29	9.278	1%	11.400	2%
30	2.182	1%	4.900	3%
31	-216	0%	400	1%
32	838	0%	800	1%
33	3.899	1%	8.400	4%
34	6.516	2%	4.700	3%
35	3.402	1%	4.400	2%
36	4.630	1%	12.200	4%
37	897	0%	3.000	3%
38	789	0%	2.200	3%
39	-319	0%	3.100	1%
40	13.182	4%	5.000	6%
<b>Totaal (Gt)</b>	<b>106.033</b>		<b>196.200</b>	



M<sub>1</sub>

Table with 40 columns (Corop 1-40) and 40 rows (1-40). Values are numerical data points representing the M1 dataset.

Corop = Corop volgens indeling in 1997.

centroids = op basis van coördinaten (x, y) volgens de officiële Rijksdriehoekmeting (RD) 1997.

dij = hellingbrede afstand tussen Corop-centroids in kilometers.

d<sub>int</sub> = interne Corop-afstand in kilometers. Maat voor interne afstand is de straal van de Corop-oppervlakte (voorgesteld als vierkant, 1997).

Rem: N/Ci 2000

M<sub>ii</sub>

Table with 40 columns (Corop 1-40) and 40 rows (1-40). Values are numerical data points representing the Mii dataset.

Corop = Corop volgens indeling 1993.

kern = dichtst bijgelegen punt van stad met grootste aantal inwoners in 1993 op Nederlandse wegennetwerk (2000).

dij = hellingbrede afstand tussen Corop-kernen in kilometers (m.b.v. ArcInfo).

d<sub>int</sub> = interne Corop-afstand in kilometers. Maat voor interne afstand is de straal van de Corop-oppervlakte (voorgesteld als vierkant, 1997).

Bron: Faculteit Civiele Techniek en Geowetenschappen sectie Infrastructuurplanning 2001

M III

Table with 40 columns (Corop 1-40) and 40 rows (1-40) showing numerical data for M III.

Corop = Corop volgens indeling 1993; kern = dichtst bijgelegen punt van stad met grootste aantal inwoners (1993) op Nederlandse wegennetwerk (2000); dij = minimale reisafstand tussen kernen over Nederlands wegennetwerk in kilometers (m.b.v. ArcInfo); di=j = gemiddelde interne woon-werkafstand van Corop o.b.v. Regionale arbeidsmarkten, migratie en woon-werkverkeer (Ekamper 2000).

Bron: Faculteit Civiele Techniek en Geovetenschappen, sectie Infrastructuurplanning, 2001

M IV

Table with 40 columns (Corop 1-40) and 40 rows (1-40) showing numerical data for M IV.

Corop = Corop volgens indeling 1993; kern = dichtst bijgelegen punt van stad met grootste aantal inwoners (1993) op Nederlandse wegennetwerk (2000); dij = minimale reistijd tussen kernen over Nederlands wegennetwerk in minuten (m.b.v. ArcInfo); di=j = gemiddelde interne reistijd afgeleid van gemiddelde interne reisafstand met een gemiddelde interne reistijd van 60 minuten.

Bron: Faculteit Civiele Techniek en Geovetenschappen, sectie Infrastructuurplanning, 2001

## **BIJLAGEN HOOFDSTUK 6**

### **LISTING ONTWIKKELD STELLA-MODEL**



$\text{Inkomende\_pendel}[1](t) = \text{Inkomende\_pendel}[1](t - dt) + (\text{hulp\_in\_inp}[1] - \text{hulp\_uit\_inp}[1]) * dt$   
 INIT  $\text{Inkomende\_pendel}[1] = 3000$   
 $\text{Inkomende\_pendel}[2](t) = \text{Inkomende\_pendel}[2](t - dt) + (\text{hulp\_in\_inp}[2] - \text{hulp\_uit\_inp}[2]) * dt$   
 INIT  $\text{Inkomende\_pendel}[2] = 2000$   
 $\text{Inkomende\_pendel}[3](t) = \text{Inkomende\_pendel}[3](t - dt) + (\text{hulp\_in\_inp}[3] - \text{hulp\_uit\_inp}[3]) * dt$   
 INIT  $\text{Inkomende\_pendel}[3] = 14000$   
 $\text{Inkomende\_pendel}[4](t) = \text{Inkomende\_pendel}[4](t - dt) + (\text{hulp\_in\_inp}[4] - \text{hulp\_uit\_inp}[4]) * dt$   
 INIT  $\text{Inkomende\_pendel}[4] = 4000$   
 $\text{Inkomende\_pendel}[5](t) = \text{Inkomende\_pendel}[5](t - dt) + (\text{hulp\_in\_inp}[5] - \text{hulp\_uit\_inp}[5]) * dt$   
 INIT  $\text{Inkomende\_pendel}[5] = 1000$   
 $\text{Inkomende\_pendel}[6](t) = \text{Inkomende\_pendel}[6](t - dt) + (\text{hulp\_in\_inp}[6] - \text{hulp\_uit\_inp}[6]) * dt$   
 INIT  $\text{Inkomende\_pendel}[6] = 3000$   
 $\text{Inkomende\_pendel}[7](t) = \text{Inkomende\_pendel}[7](t - dt) + (\text{hulp\_in\_inp}[7] - \text{hulp\_uit\_inp}[7]) * dt$   
 INIT  $\text{Inkomende\_pendel}[7] = 4000$   
 $\text{Inkomende\_pendel}[8](t) = \text{Inkomende\_pendel}[8](t - dt) + (\text{hulp\_in\_inp}[8] - \text{hulp\_uit\_inp}[8]) * dt$   
 INIT  $\text{Inkomende\_pendel}[8] = 2000$   
 $\text{Inkomende\_pendel}[9](t) = \text{Inkomende\_pendel}[9](t - dt) + (\text{hulp\_in\_inp}[9] - \text{hulp\_uit\_inp}[9]) * dt$   
 INIT  $\text{Inkomende\_pendel}[9] = 3000$   
 $\text{Inkomende\_pendel}[10](t) = \text{Inkomende\_pendel}[10](t - dt) + (\text{hulp\_in\_inp}[10] - \text{hulp\_uit\_inp}[10]) * dt$   
 INIT  $\text{Inkomende\_pendel}[10] = 7000$   
 $\text{Inkomende\_pendel}[11](t) = \text{Inkomende\_pendel}[11](t - dt) + (\text{hulp\_in\_inp}[11] - \text{hulp\_uit\_inp}[11]) * dt$   
 INIT  $\text{Inkomende\_pendel}[11] = 6000$   
 $\text{Inkomende\_pendel}[12](t) = \text{Inkomende\_pendel}[12](t - dt) + (\text{hulp\_in\_inp}[12] - \text{hulp\_uit\_inp}[12]) * dt$   
 INIT  $\text{Inkomende\_pendel}[12] = 5000$   
 $\text{Inkomende\_pendel}[13](t) = \text{Inkomende\_pendel}[13](t - dt) + (\text{hulp\_in\_inp}[13] - \text{hulp\_uit\_inp}[13]) * dt$   
 INIT  $\text{Inkomende\_pendel}[13] = 16000$   
 $\text{Inkomende\_pendel}[14](t) = \text{Inkomende\_pendel}[14](t - dt) + (\text{hulp\_in\_inp}[14] - \text{hulp\_uit\_inp}[14]) * dt$   
 INIT  $\text{Inkomende\_pendel}[14] = 6000$   
 $\text{Inkomende\_pendel}[15](t) = \text{Inkomende\_pendel}[15](t - dt) + (\text{hulp\_in\_inp}[15] - \text{hulp\_uit\_inp}[15]) * dt$   
 INIT  $\text{Inkomende\_pendel}[15] = 15000$   
 $\text{Inkomende\_pendel}[16](t) = \text{Inkomende\_pendel}[16](t - dt) + (\text{hulp\_in\_inp}[16] - \text{hulp\_uit\_inp}[16]) * dt$   
 INIT  $\text{Inkomende\_pendel}[16] = 5000$   
 $\text{Inkomende\_pendel}[17](t) = \text{Inkomende\_pendel}[17](t - dt) + (\text{hulp\_in\_inp}[17] - \text{hulp\_uit\_inp}[17]) * dt$   
 INIT  $\text{Inkomende\_pendel}[17] = 31000$   
 $\text{Inkomende\_pendel}[18](t) = \text{Inkomende\_pendel}[18](t - dt) + (\text{hulp\_in\_inp}[18] - \text{hulp\_uit\_inp}[18]) * dt$   
 INIT  $\text{Inkomende\_pendel}[18] = 3000$   
 $\text{Inkomende\_pendel}[19](t) = \text{Inkomende\_pendel}[19](t - dt) + (\text{hulp\_in\_inp}[19] - \text{hulp\_uit\_inp}[19]) * dt$   
 INIT  $\text{Inkomende\_pendel}[19] = 6000$   
 $\text{Inkomende\_pendel}[20](t) = \text{Inkomende\_pendel}[20](t - dt) + (\text{hulp\_in\_inp}[20] - \text{hulp\_uit\_inp}[20]) * dt$   
 INIT  $\text{Inkomende\_pendel}[20] = 16000$   
 $\text{Inkomende\_pendel}[21](t) = \text{Inkomende\_pendel}[21](t - dt) + (\text{hulp\_in\_inp}[21] - \text{hulp\_uit\_inp}[21]) * dt$   
 INIT  $\text{Inkomende\_pendel}[21] = 12000$

$$\text{Inkomende\_pendel}[22](t) = \text{Inkomende\_pendel}[22](t - dt) + (\text{hulp\_in\_inp}[22] - \text{hulp\_uit\_inp}[22]) * dt$$
 INIT Inkomende\_pendel[22] = 9000  

$$\text{Inkomende\_pendel}[23](t) = \text{Inkomende\_pendel}[23](t - dt) + (\text{hulp\_in\_inp}[23] - \text{hulp\_uit\_inp}[23]) * dt$$
 INIT Inkomende\_pendel[23] = 78000  

$$\text{Inkomende\_pendel}[24](t) = \text{Inkomende\_pendel}[24](t - dt) + (\text{hulp\_in\_inp}[24] - \text{hulp\_uit\_inp}[24]) * dt$$
 INIT Inkomende\_pendel[24] = 13000  

$$\text{Inkomende\_pendel}[25](t) = \text{Inkomende\_pendel}[25](t - dt) + (\text{hulp\_in\_inp}[25] - \text{hulp\_uit\_inp}[25]) * dt$$
 INIT Inkomende\_pendel[25] = 10000  

$$\text{Inkomende\_pendel}[26](t) = \text{Inkomende\_pendel}[26](t - dt) + (\text{hulp\_in\_inp}[26] - \text{hulp\_uit\_inp}[26]) * dt$$
 INIT Inkomende\_pendel[26] = 43000  

$$\text{Inkomende\_pendel}[27](t) = \text{Inkomende\_pendel}[27](t - dt) + (\text{hulp\_in\_inp}[27] - \text{hulp\_uit\_inp}[27]) * dt$$
 INIT Inkomende\_pendel[27] = 13000  

$$\text{Inkomende\_pendel}[28](t) = \text{Inkomende\_pendel}[28](t - dt) + (\text{hulp\_in\_inp}[28] - \text{hulp\_uit\_inp}[28]) * dt$$
 INIT Inkomende\_pendel[28] = 9000  

$$\text{Inkomende\_pendel}[29](t) = \text{Inkomende\_pendel}[29](t - dt) + (\text{hulp\_in\_inp}[29] - \text{hulp\_uit\_inp}[29]) * dt$$
 INIT Inkomende\_pendel[29] = 49000  

$$\text{Inkomende\_pendel}[30](t) = \text{Inkomende\_pendel}[30](t - dt) + (\text{hulp\_in\_inp}[30] - \text{hulp\_uit\_inp}[30]) * dt$$
 INIT Inkomende\_pendel[30] = 13000  

$$\text{Inkomende\_pendel}[31](t) = \text{Inkomende\_pendel}[31](t - dt) + (\text{hulp\_in\_inp}[31] - \text{hulp\_uit\_inp}[31]) * dt$$
 INIT Inkomende\_pendel[31] = 3000  

$$\text{Inkomende\_pendel}[32](t) = \text{Inkomende\_pendel}[32](t - dt) + (\text{hulp\_in\_inp}[32] - \text{hulp\_uit\_inp}[32]) * dt$$
 INIT Inkomende\_pendel[32] = 2000  

$$\text{Inkomende\_pendel}[33](t) = \text{Inkomende\_pendel}[33](t - dt) + (\text{hulp\_in\_inp}[33] - \text{hulp\_uit\_inp}[33]) * dt$$
 INIT Inkomende\_pendel[33] = 11000  

$$\text{Inkomende\_pendel}[34](t) = \text{Inkomende\_pendel}[34](t - dt) + (\text{hulp\_in\_inp}[34] - \text{hulp\_uit\_inp}[34]) * dt$$
 INIT Inkomende\_pendel[34] = 10000  

$$\text{Inkomende\_pendel}[35](t) = \text{Inkomende\_pendel}[35](t - dt) + (\text{hulp\_in\_inp}[35] - \text{hulp\_uit\_inp}[35]) * dt$$
 INIT Inkomende\_pendel[35] = 12000  

$$\text{Inkomende\_pendel}[36](t) = \text{Inkomende\_pendel}[36](t - dt) + (\text{hulp\_in\_inp}[36] - \text{hulp\_uit\_inp}[36]) * dt$$
 INIT Inkomende\_pendel[36] = 19000  

$$\text{Inkomende\_pendel}[37](t) = \text{Inkomende\_pendel}[37](t - dt) + (\text{hulp\_in\_inp}[37] - \text{hulp\_uit\_inp}[37]) * dt$$
 INIT Inkomende\_pendel[37] = 6000  

$$\text{Inkomende\_pendel}[38](t) = \text{Inkomende\_pendel}[38](t - dt) + (\text{hulp\_in\_inp}[38] - \text{hulp\_uit\_inp}[38]) * dt$$
 INIT Inkomende\_pendel[38] = 5000  

$$\text{Inkomende\_pendel}[39](t) = \text{Inkomende\_pendel}[39](t - dt) + (\text{hulp\_in\_inp}[39] - \text{hulp\_uit\_inp}[39]) * dt$$
 INIT Inkomende\_pendel[39] = 10000

```

Inkomende_pendel[40](t) = Inkomende_pendel[40](t - dt) + (hulp_in_inp[40] - hulp_uit_inp[40])
* dt
INIT Inkomende_pendel[40] = 2000
hulp_in_inp[van] = INP[van]+grens_INP[van]
hulp_uit_inp[van] = Inkomende_pendel[van]
Uitgaande_pendel[1](t) = Uitgaande_pendel[1](t - dt) + (hulp_in_uitp[1] - hulp_uit_uitp[1]) * dt
INIT Uitgaande_pendel[1] = 5000
Uitgaande_pendel[2](t) = Uitgaande_pendel[2](t - dt) + (hulp_in_uitp[2] - hulp_uit_uitp[2]) * dt
INIT Uitgaande_pendel[2] = 2000
Uitgaande_pendel[3](t) = Uitgaande_pendel[3](t - dt) + (hulp_in_uitp[3] - hulp_uit_uitp[3]) * dt
INIT Uitgaande_pendel[3] = 5000
Uitgaande_pendel[4](t) = Uitgaande_pendel[4](t - dt) + (hulp_in_uitp[4] - hulp_uit_uitp[4]) * dt
INIT Uitgaande_pendel[4] = 4000
Uitgaande_pendel[5](t) = Uitgaande_pendel[5](t - dt) + (hulp_in_uitp[5] - hulp_uit_uitp[5]) * dt
INIT Uitgaande_pendel[5] = 3000
Uitgaande_pendel[6](t) = Uitgaande_pendel[6](t - dt) + (hulp_in_uitp[6] - hulp_uit_uitp[6]) * dt
INIT Uitgaande_pendel[6] = 4000
Uitgaande_pendel[7](t) = Uitgaande_pendel[7](t - dt) + (hulp_in_uitp[7] - hulp_uit_uitp[7]) * dt
INIT Uitgaande_pendel[7] = 9000
Uitgaande_pendel[8](t) = Uitgaande_pendel[8](t - dt) + (hulp_in_uitp[8] - hulp_uit_uitp[8]) * dt
INIT Uitgaande_pendel[8] = 3000
Uitgaande_pendel[9](t) = Uitgaande_pendel[9](t - dt) + (hulp_in_uitp[9] - hulp_uit_uitp[9]) * dt
INIT Uitgaande_pendel[9] = 3000
Uitgaande_pendel[10](t) = Uitgaande_pendel[10](t - dt) + (hulp_in_uitp[10] - hulp_uit_uitp[10])
* dt
INIT Uitgaande_pendel[10] = 9000
Uitgaande_pendel[11](t) = Uitgaande_pendel[11](t - dt) + (hulp_in_uitp[11] - hulp_uit_uitp[11])
* dt
INIT Uitgaande_pendel[11] = 4000
Uitgaande_pendel[12](t) = Uitgaande_pendel[12](t - dt) + (hulp_in_uitp[12] - hulp_uit_uitp[12])
* dt
INIT Uitgaande_pendel[12] = 7000
Uitgaande_pendel[13](t) = Uitgaande_pendel[13](t - dt) + (hulp_in_uitp[13] - hulp_uit_uitp[13])
* dt
INIT Uitgaande_pendel[13] = 20000
Uitgaande_pendel[14](t) = Uitgaande_pendel[14](t - dt) + (hulp_in_uitp[14] - hulp_uit_uitp[14])
* dt
INIT Uitgaande_pendel[14] = 11000
Uitgaande_pendel[15](t) = Uitgaande_pendel[15](t - dt) + (hulp_in_uitp[15] - hulp_uit_uitp[15])
* dt
INIT Uitgaande_pendel[15] = 12000
Uitgaande_pendel[16](t) = Uitgaande_pendel[16](t - dt) + (hulp_in_uitp[16] - hulp_uit_uitp[16])
* dt
INIT Uitgaande_pendel[16] = 12000
Uitgaande_pendel[17](t) = Uitgaande_pendel[17](t - dt) + (hulp_in_uitp[17] - hulp_uit_uitp[17])
* dt
INIT Uitgaande_pendel[17] = 31000
Uitgaande_pendel[18](t) = Uitgaande_pendel[18](t - dt) + (hulp_in_uitp[18] - hulp_uit_uitp[18])
* dt
INIT Uitgaande_pendel[18] = 15000
Uitgaande_pendel[19](t) = Uitgaande_pendel[19](t - dt) + (hulp_in_uitp[19] - hulp_uit_uitp[19])
* dt
INIT Uitgaande_pendel[19] = 16000
Uitgaande_pendel[20](t) = Uitgaande_pendel[20](t - dt) + (hulp_in_uitp[20] - hulp_uit_uitp[20])
* dt

```

```

INIT Uitgaande_pendel[20] = 12000
Uitgaande_pendel[21](t) = Uitgaande_pendel[21](t - dt) + (hulp_in_uitp[21] - hulp_uit_uitp[21])
* dt
INIT Uitgaande_pendel[21] = 22000
Uitgaande_pendel[22](t) = Uitgaande_pendel[22](t - dt) + (hulp_in_uitp[22] - hulp_uit_uitp[22])
* dt
INIT Uitgaande_pendel[22] = 12000
Uitgaande_pendel[23](t) = Uitgaande_pendel[23](t - dt) + (hulp_in_uitp[23] - hulp_uit_uitp[23])
* dt
INIT Uitgaande_pendel[23] = 60000
Uitgaande_pendel[24](t) = Uitgaande_pendel[24](t - dt) + (hulp_in_uitp[24] - hulp_uit_uitp[24])
* dt
INIT Uitgaande_pendel[24] = 17000
Uitgaande_pendel[25](t) = Uitgaande_pendel[25](t - dt) + (hulp_in_uitp[25] - hulp_uit_uitp[25])
* dt
INIT Uitgaande_pendel[25] = 24000
Uitgaande_pendel[26](t) = Uitgaande_pendel[26](t - dt) + (hulp_in_uitp[26] - hulp_uit_uitp[26])
* dt
INIT Uitgaande_pendel[26] = 20000
Uitgaande_pendel[27](t) = Uitgaande_pendel[27](t - dt) + (hulp_in_uitp[27] - hulp_uit_uitp[27])
* dt
INIT Uitgaande_pendel[27] = 16000
Uitgaande_pendel[28](t) = Uitgaande_pendel[28](t - dt) + (hulp_in_uitp[28] - hulp_uit_uitp[28])
* dt
INIT Uitgaande_pendel[28] = 24000
Uitgaande_pendel[29](t) = Uitgaande_pendel[29](t - dt) + (hulp_in_uitp[29] - hulp_uit_uitp[29])
* dt
INIT Uitgaande_pendel[29] = 23000
Uitgaande_pendel[30](t) = Uitgaande_pendel[30](t - dt) + (hulp_in_uitp[30] - hulp_uit_uitp[30])
* dt
INIT Uitgaande_pendel[30] = 26000
Uitgaande_pendel[31](t) = Uitgaande_pendel[31](t - dt) + (hulp_in_uitp[31] - hulp_uit_uitp[31])
* dt
INIT Uitgaande_pendel[31] = 2000
Uitgaande_pendel[32](t) = Uitgaande_pendel[32](t - dt) + (hulp_in_uitp[32] - hulp_uit_uitp[32])
* dt
INIT Uitgaande_pendel[32] = 5000
Uitgaande_pendel[33](t) = Uitgaande_pendel[33](t - dt) + (hulp_in_uitp[33] - hulp_uit_uitp[33])
* dt
INIT Uitgaande_pendel[33] = 20000
Uitgaande_pendel[34](t) = Uitgaande_pendel[34](t - dt) + (hulp_in_uitp[34] - hulp_uit_uitp[34])
* dt
INIT Uitgaande_pendel[34] = 17000
Uitgaande_pendel[35](t) = Uitgaande_pendel[35](t - dt) + (hulp_in_uitp[35] - hulp_uit_uitp[35])
* dt
INIT Uitgaande_pendel[35] = 15000
Uitgaande_pendel[36](t) = Uitgaande_pendel[36](t - dt) + (hulp_in_uitp[36] - hulp_uit_uitp[36])
* dt
INIT Uitgaande_pendel[36] = 9000
Uitgaande_pendel[37](t) = Uitgaande_pendel[37](t - dt) + (hulp_in_uitp[37] - hulp_uit_uitp[37])
* dt
INIT Uitgaande_pendel[37] = 7000
Uitgaande_pendel[38](t) = Uitgaande_pendel[38](t - dt) + (hulp_in_uitp[38] - hulp_uit_uitp[38])
* dt
INIT Uitgaande_pendel[38] = 8000

```



$$\text{Uitgaande\_pendel}[39](t) = \text{Uitgaande\_pendel}[39](t - dt) + (\text{hulp\_in\_uitp}[39] - \text{hulp\_uit\_uitp}[39]) * dt$$

$$\text{INIT Uitgaande\_pendel}[39] = 13000$$

$$\text{Uitgaande\_pendel}[40](t) = \text{Uitgaande\_pendel}[40](t - dt) + (\text{hulp\_in\_uitp}[40] - \text{hulp\_uit\_uitp}[40]) * dt$$

$$\text{INIT Uitgaande\_pendel}[40] = 2000$$

$$\text{hulp\_in\_uitp}[van] = \text{UITP}[van] + \text{grens\_UITP}[van]$$

$$\text{hulp\_uit\_uitp}[van] = \text{Uitgaande\_pendel}[van]$$

$$\text{arbeidsplaatsen}[van] = \text{BEDRIJFSVESTIGINGEN}[van] * \text{banen\_per\_BV}[van] * \text{corr\_banen}[van] * \text{afwijking\_VT}[van] * \text{conjuncturele\_ontwikkeling}$$

$$\text{banen\_per\_BV}[1] = 6.6$$

$$\text{banen\_per\_BV}[2] = 7.9$$

$$\text{banen\_per\_BV}[3] = 9.3$$

$$\text{banen\_per\_BV}[4] = 7.3$$

$$\text{banen\_per\_BV}[5] = 5$$

$$\text{banen\_per\_BV}[6] = 6.4$$

$$\text{banen\_per\_BV}[7] = 8.1$$

$$\text{banen\_per\_BV}[8] = 7.5$$

$$\text{banen\_per\_BV}[9] = 5.8$$

$$\text{banen\_per\_BV}[10] = 7.3$$

$$\text{banen\_per\_BV}[11] = 8.6$$

$$\text{banen\_per\_BV}[12] = 7.8$$

$$\text{banen\_per\_BV}[13] = 7.7$$

$$\text{banen\_per\_BV}[14] = 6.2$$

$$\text{banen\_per\_BV}[15] = 9.6$$

$$\text{banen\_per\_BV}[16] = 5.0$$

$$\text{banen\_per\_BV}[17] = 9.6$$

$$\text{banen\_per\_BV}[18] = 5.5$$

$$\text{banen\_per\_BV}[19] = 7.3$$

$$\text{banen\_per\_BV}[20] = 8.6$$

$$\text{banen\_per\_BV}[21] = 8.3$$

$$\text{banen\_per\_BV}[22] = 9.7$$

$$\text{banen\_per\_BV}[23] = 9.7$$

$$\text{banen\_per\_BV}[24] = 7.2$$

$$\text{banen\_per\_BV}[25] = 8.4$$

$$\text{banen\_per\_BV}[26] = 11.1$$

$$\text{banen\_per\_BV}[27] = 7.7$$

$$\text{banen\_per\_BV}[28] = 6.6$$

$$\text{banen\_per\_BV}[29] = 9.4$$

$$\text{banen\_per\_BV}[30] = 8.5$$

$$\text{banen\_per\_BV}[31] = 6$$

$$\text{banen\_per\_BV}[32] = 6.4$$

$$\text{banen\_per\_BV}[33] = 7.8$$

$$\text{banen\_per\_BV}[34] = 7.3$$

$$\text{banen\_per\_BV}[35] = 8.2$$

$$\text{banen\_per\_BV}[36] = 8.5$$

$$\text{banen\_per\_BV}[37] = 7.6$$

$$\text{banen\_per\_BV}[38] = 6.9$$

$$\text{banen\_per\_BV}[39] = 9.3$$

$$\text{banen\_per\_BV}[40] = 6.3$$

$$\text{BBAR}[van] = (\text{beroepsbevolking\_werkend}[van]) / \text{arbeidsplaatsen}[van]$$

$$\text{beroepsbevolking\_werkend}[van] = (\text{beroepsbevolking\_wonend}[van] + \text{pendelsaldo}[van])$$

$$\text{beroepsbevolking\_wonend}[van] = \text{pot\_beroepsbevolking}[van] * \text{bruto\_beroeps participatie}[van] * \text{trend\_beroeps participatie}[van]$$

bruto\_beroeps participatie[1] = 0.47  
bruto\_beroeps participatie[2] = 0.46  
bruto\_beroeps participatie[3] = 0.49  
bruto\_beroeps participatie[4] = 0.48  
bruto\_beroeps participatie[5] = 0.5  
bruto\_beroeps participatie[6] = 0.48  
bruto\_beroeps participatie[7] = 0.48  
bruto\_beroeps participatie[8] = 0.49  
bruto\_beroeps participatie[9] = 0.48  
bruto\_beroeps participatie[10] = 0.5  
bruto\_beroeps participatie[11] = 0.51  
bruto\_beroeps participatie[12] = 0.47  
bruto\_beroeps participatie[13] = 0.49  
bruto\_beroeps participatie[14] = 0.52  
bruto\_beroeps participatie[15] = 0.51  
bruto\_beroeps participatie[16] = 0.51  
bruto\_beroeps participatie[17] = 0.52  
bruto\_beroeps participatie[18] = 0.52  
bruto\_beroeps participatie[19] = 0.51  
bruto\_beroeps participatie[20] = 0.5  
bruto\_beroeps participatie[21] = 0.54  
bruto\_beroeps participatie[22] = 0.53  
bruto\_beroeps participatie[23] = 0.45  
bruto\_beroeps participatie[24] = 0.53  
bruto\_beroeps participatie[25] = 0.53  
bruto\_beroeps participatie[26] = 0.53  
bruto\_beroeps participatie[27] = 0.53  
bruto\_beroeps participatie[28] = 0.53  
bruto\_beroeps participatie[29] = 0.49  
bruto\_beroeps participatie[30] = 0.5  
bruto\_beroeps participatie[31] = 0.51  
bruto\_beroeps participatie[32] = 0.49  
bruto\_beroeps participatie[33] = 0.51  
bruto\_beroeps participatie[34] = 0.5  
bruto\_beroeps participatie[35] = 0.51  
bruto\_beroeps participatie[36] = 0.51  
bruto\_beroeps participatie[37] = 0.53  
bruto\_beroeps participatie[38] = 0.49  
bruto\_beroeps participatie[39] = 0.48  
bruto\_beroeps participatie[40] = 0.53  
corr\_banen[van] = TIME  
grens\_INP[van] = arbeidsplaatsen[van]\*pct\_grensinp[van]\*ABM\_grensinpendel[van]  
grens\_UITP[van] = arbeidsplaatsen[van]\*pct\_grensuitp[van]\*ABM\_grensuitpendel[van]  
pct\_grensinp[1] = 0  
pct\_grensinp[2] = 0  
pct\_grensinp[3] = 0  
pct\_grensinp[4] = 0  
pct\_grensinp[5] = 0  
pct\_grensinp[6] = 0  
pct\_grensinp[7] = 0  
pct\_grensinp[8] = 0.001  
pct\_grensinp[9] = 0.001  
pct\_grensinp[10] = 0.001  
pct\_grensinp[11] = 0  
pct\_grensinp[12] = 0.001

pct\_grensinp[13] = 0  
pct\_grensinp[14] = 0.001  
pct\_grensinp[15] = 0.001  
pct\_grensinp[16] = 0  
pct\_grensinp[17] = 0  
pct\_grensinp[18] = 0  
pct\_grensinp[19] = 0  
pct\_grensinp[20] = 0  
pct\_grensinp[21] = 0  
pct\_grensinp[22] = 0  
pct\_grensinp[23] = 0  
pct\_grensinp[24] = 0  
pct\_grensinp[25] = 0  
pct\_grensinp[26] = 0  
pct\_grensinp[27] = 0  
pct\_grensinp[28] = 0  
pct\_grensinp[29] = 0  
pct\_grensinp[30] = 0  
pct\_grensinp[31] = 0.038  
pct\_grensinp[32] = 0.001  
pct\_grensinp[33] = 0.007  
pct\_grensinp[34] = 0.007  
pct\_grensinp[35] = 0.002  
pct\_grensinp[36] = 0.014  
pct\_grensinp[37] = 0.004  
pct\_grensinp[38] = 0.018  
pct\_grensinp[39] = 0.025  
pct\_grensinp[40] = 0  
pct\_grensuitp[1] = 0.003  
pct\_grensuitp[2] = 0.001  
pct\_grensuitp[3] = 0  
pct\_grensuitp[4] = 0  
pct\_grensuitp[5] = 0  
pct\_grensuitp[6] = 0  
pct\_grensuitp[7] = 0  
pct\_grensuitp[8] = 0.005  
pct\_grensuitp[9] = 0  
pct\_grensuitp[10] = 0  
pct\_grensuitp[11] = 0  
pct\_grensuitp[12] = 0.006  
pct\_grensuitp[13] = 0  
pct\_grensuitp[14] = 0.013  
pct\_grensuitp[15] = 0.004  
pct\_grensuitp[16] = 0  
pct\_grensuitp[17] = 0  
pct\_grensuitp[18] = 0  
pct\_grensuitp[19] = 0  
pct\_grensuitp[20] = 0  
pct\_grensuitp[21] = 0  
pct\_grensuitp[22] = 0  
pct\_grensuitp[23] = 0  
pct\_grensuitp[24] = 0  
pct\_grensuitp[25] = 0  
pct\_grensuitp[26] = 0  
pct\_grensuitp[27] = 0

pct\_grensuitp[28] = 0  
pct\_grensuitp[29] = 0  
pct\_grensuitp[30] = 0  
pct\_grensuitp[31] = 0.037  
pct\_grensuitp[32] = 0.003  
pct\_grensuitp[33] = 0.007  
pct\_grensuitp[34] = 0.002  
pct\_grensuitp[35] = 0.001  
pct\_grensuitp[36] = 0.003  
pct\_grensuitp[37] = 0.023  
pct\_grensuitp[38] = 0.018  
pct\_grensuitp[39] = 0.033  
pct\_grensuitp[40] = 0  
pct\_inarb[1] = 0.6  
pct\_inarb[2] = 0.45  
pct\_inarb[3] = 0.6  
pct\_inarb[4] = 0.3  
pct\_inarb[5] = 0.3  
pct\_inarb[6] = 0.2  
pct\_inarb[7] = 0.3  
pct\_inarb[8] = 0.3  
pct\_inarb[9] = 0.2  
pct\_inarb[10] = 0.3  
pct\_inarb[11] = 0.3  
pct\_inarb[12] = 0.15  
pct\_inarb[13] = 0.3  
pct\_inarb[14] = 0.25  
pct\_inarb[15] = 0.2  
pct\_inarb[16] = 0.3  
pct\_inarb[17] = 0.15  
pct\_inarb[18] = 0.3  
pct\_inarb[19] = 0.3  
pct\_inarb[20] = 0.3  
pct\_inarb[21] = 0.1  
pct\_inarb[22] = 0.2  
pct\_inarb[23] = 0.7  
pct\_inarb[24] = 0.3  
pct\_inarb[25] = 0.2  
pct\_inarb[26] = 0.2  
pct\_inarb[27] = 0.25  
pct\_inarb[28] = 0.3  
pct\_inarb[29] = 0.3  
pct\_inarb[30] = 0.3  
pct\_inarb[31] = 0.3  
pct\_inarb[32] = 0.3  
pct\_inarb[33] = 0.25  
pct\_inarb[34] = 0.2  
pct\_inarb[35] = 0.3  
pct\_inarb[36] = 0.3  
pct\_inarb[37] = 0.3  
pct\_inarb[38] = 0.1  
pct\_inarb[39] = 0.3  
pct\_inarb[40] = 0.3  
pct\_UITARB[1] = 0.3  
pct\_UITARB[2] = 0.2

pct\_UITARB[3] = 0.25  
 pct\_UITARB[4] = 0.3  
 pct\_UITARB[5] = 0.3  
 pct\_UITARB[6] = 0.3  
 pct\_UITARB[7] = 0.45  
 pct\_UITARB[8] = 0.2  
 pct\_UITARB[9] = 0.25  
 pct\_UITARB[10] = 0.25  
 pct\_UITARB[11] = 0.25  
 pct\_UITARB[12] = 0.1  
 pct\_UITARB[13] = 0.25  
 pct\_UITARB[14] = 0.55  
 pct\_UITARB[15] = 0.23  
 pct\_UITARB[16] = 0.35  
 pct\_UITARB[17] = 0.23  
 pct\_UITARB[18] = 0.45  
 pct\_UITARB[19] = 0.6  
 pct\_UITARB[20] = 0.3  
 pct\_UITARB[21] = 0.5  
 pct\_UITARB[22] = 0.35  
 pct\_UITARB[23] = 0.6  
 pct\_UITARB[24] = 0.35  
 pct\_UITARB[25] = 0.45  
 pct\_UITARB[26] = 0.25  
 pct\_UITARB[27] = 0.35  
 pct\_UITARB[28] = 0.5  
 pct\_UITARB[29] = 0.45  
 pct\_UITARB[30] = 0.45  
 pct\_UITARB[31] = 0.3  
 pct\_UITARB[32] = 0.45  
 pct\_UITARB[33] = 0.25  
 pct\_UITARB[34] = 0.35  
 pct\_UITARB[35] = 0.35  
 pct\_UITARB[36] = 0.15  
 pct\_UITARB[37] = 0.25  
 pct\_UITARB[38] = 0.25  
 pct\_UITARB[39] = 0.25  
 pct\_UITARB[40] = 0.35  
 pendelsaldo[van] = Inkomende\_pendel[van]+Uitgaande\_pendel[van]  
 pot\_beroepsbevolking[van] = BEVOLKINfG[van]\*pct\_pot\_bb[van]  
 pot\_INARB[van] = arbeidsplaatsen[van]\*ABM\_INARB[van]\*pct\_inarb[van]  
 pot\_UITARB[van] = beroepsbevolking\_wonend[van]\*pct\_UITARB[van]\*ABM\_UITARB[van]  
 ABBM\_EM\_BV[van] = GRAPH(BBAR[van])  
 (0.00, 1.20), (0.2, 1.20), (0.4, 1.19), (0.6, 1.14), (0.8, 1.09), (1.00, 1.00), (1.20, 0.91), (1.40, 0.85),  
 (1.60, 0.81), (1.80, 0.8), (2.00, 0.79)  
 ABBM\_IMM\_BV[van] = GRAPH(BBAR[van])  
 (0.00, 0.79), (0.2, 0.79), (0.4, 0.81), (0.6, 0.85), (0.8, 0.92), (1.00, 1.00), (1.20, 1.07), (1.40, 1.12),  
 (1.60, 1.16), (1.80, 1.18), (2.00, 1.19)  
 ABBM\_INM\_BV[van] = GRAPH(BBAR[van])  
 (0.00, 0.42), (0.2, 0.44), (0.4, 0.51), (0.6, 0.63), (0.8, 0.78), (1.00, 1.00), (1.20, 1.24), (1.40, 1.50),  
 (1.60, 1.62), (1.80, 1.69), (2.00, 1.70)  
 ABBM\_UITM\_BV[van] = GRAPH(BBAR[van])  
 (0.00, 1.60), (0.2, 1.59), (0.4, 1.57), (0.6, 1.47), (0.8, 1.29), (1.00, 1.00), (1.20, 0.72), (1.40, 0.53),  
 (1.60, 0.43), (1.80, 0.41), (2.00, 0.4)  
 ABM\_EM\_BEV[van] = GRAPH(BBAR[van])

(0.00, 0.75), (0.2, 0.75), (0.4, 0.77), (0.6, 0.82), (0.8, 0.9), (1.00, 1.00), (1.20, 1.11), (1.40, 1.18), (1.60, 1.24), (1.80, 1.26), (2.00, 1.28)

ABM\_grensinpendel[van] = GRAPH(BBAR[van])  
(0.00, 1.16), (0.2, 1.16), (0.4, 1.15), (0.6, 1.12), (0.8, 1.07), (1.00, 1.00), (1.20, 0.96), (1.40, 0.92), (1.60, 0.89), (1.80, 0.87), (2.00, 0.86)

ABM\_grensuitpendel[van] = GRAPH(BBAR[van])  
(0.00, 0.83), (0.2, 0.83), (0.4, 0.84), (0.6, 0.87), (0.8, 0.92), (1.00, 1.00), (1.20, 1.07), (1.40, 1.11), (1.60, 1.14), (1.80, 1.15), (2.00, 1.15)

ABM\_IMM\_BEV[van] = GRAPH(BBAR[van])  
(0.00, 1.60), (0.2, 1.59), (0.4, 1.57), (0.6, 1.47), (0.8, 1.29), (1.00, 1.00), (1.20, 0.72), (1.40, 0.53), (1.60, 0.43), (1.80, 0.41), (2.00, 0.4)

ABM\_INARB[van] = GRAPH(BBAR[van])  
(0.00, 1.65), (0.2, 1.64), (0.4, 1.62), (0.6, 1.55), (0.8, 1.39), (1.00, 1.00), (1.20, 0.64), (1.40, 0.5), (1.60, 0.42), (1.80, 0.38), (2.00, 0.36)

ABM\_UITARB[van] = GRAPH(BBAR[van])  
(0.00, 0.7), (0.2, 0.7), (0.4, 0.72), (0.6, 0.77), (0.8, 0.86), (1.00, 1.00), (1.20, 1.14), (1.40, 1.23), (1.60, 1.28), (1.80, 1.30), (2.00, 1.31)

afwijking\_VT[van] = GRAPH(TIME)  
(1970, 1.00), (2030, 1.35)

conjuncturele\_ontwikkeling = GRAPH(TIME)  
(1970, 0.989), (1971, 0.98), (1972, 0.991), (1973, 1.01), (1974, 1.02), (1975, 1.01), (1976, 0.99), (1977, 0.98), (1978, 0.991), (1979, 1.01), (1980, 1.02), (1981, 1.01), (1982, 0.99), (1983, 0.98), (1984, 0.991), (1985, 1.01), (1986, 1.02), (1987, 1.01), (1988, 0.99), (1989, 0.98), (1990, 0.991), (1991, 1.01), (1992, 1.02), (1993, 1.01), (1994, 0.99), (1995, 0.98), (1996, 0.991), (1997, 1.01), (1998, 1.02), (1999, 1.01), (2000, 0.99), (2001, 0.98), (2002, 0.991), (2003, 1.01), (2004, 1.02), (2005, 1.01), (2006, 0.99), (2007, 0.98), (2008, 0.991), (2009, 1.01), (2010, 1.02), (2011, 1.01), (2012, 0.99), (2013, 0.98), (2014, 0.991), (2015, 1.01), (2016, 1.02), (2017, 1.01), (2018, 0.99), (2019, 0.98), (2020, 0.991), (2021, 1.01), (2022, 1.02), (2023, 1.01), (2024, 0.99), (2025, 0.981), (2026, 0.984), (2027, 1.01), (2028, 1.02), (2029, 1.01), (2030, 0.987)

corr\_banen[van] = TIME

trend\_beroeps participatie[van] = GRAPH(TIME)  
(1970, 1.00), (1971, 1.02), (1972, 1.04), (1973, 1.05), (1974, 1.06), (1975, 1.08), (1976, 1.09), (1977, 1.10), (1978, 1.12), (1979, 1.15), (1980, 1.18), (1981, 1.20), (1982, 1.20), (1983, 1.19), (1984, 1.17), (1985, 1.17), (1986, 1.18), (1987, 1.18), (1988, 1.19), (1989, 1.19), (1990, 1.20), (1991, 1.22), (1992, 1.24), (1993, 1.24), (1994, 1.23), (1995, 1.21), (1996, 1.22), (1997, 1.23), (1998, 1.24), (1999, 1.25), (2000, 1.26), (2001, 1.27), (2002, 1.29), (2003, 1.30), (2004, 1.30), (2005, 1.32), (2006, 1.32), (2007, 1.33), (2008, 1.33), (2009, 1.33), (2010, 1.33)

BEDRIJFSRUIMTEVOORRAAD[van](t) = BEDRIJFSRUIMTEVOORRAAD[van](t - dt) + (Bedrijfsruimtevermeerdering[van] - Bedrijfsruimtevermindering[van]) \* dt

INIT BEDRIJFSRUIMTEVOORRAAD[van] = BEDRIJFSVESTIGINGEN[van]

Bedrijfsruimtevermeerdering[van] =  
BEDRIJFSRUIMTEVOORRAAD[van]\*bedrijfsruimtebouwfactor[van]

Bedrijfsruimtevermindering[van] =  
BEDRIJFSRUIMTEVOORRAAD[van]\*bedrijfsruimtesloopfactor[van]

bedrijfsruimtebouwfactor[van] =  
pct\_BEDRRUIMVERMEER[van]\*BRUIMBVBM[van]\*BRUIMLANDBM[van]

bedrijfsruimtesloopfactor[van] =  
pct\_BEDRRUIMVERMIN[van]\*BRUIMBVSM[van]\*BRUIMLANDSM[van]

BVBR[van] = BEDRIJFSVESTIGINGEN[van]/BEDRIJFSRUIMTEVOORRAAD[van]

pct\_BEDRRUIMVERMEER[1] = 0.015  
pct\_BEDRRUIMVERMEER[2] = 0.02  
pct\_BEDRRUIMVERMEER[3] = 0.015  
pct\_BEDRRUIMVERMEER[4] = 0.02  
pct\_BEDRRUIMVERMEER[5] = 0.02  
pct\_BEDRRUIMVERMEER[6] = 0.02

pct\_BEDRRUIMVERMEER[7] = 0.02  
 pct\_BEDRRUIMVERMEER[8] = 0.02  
 pct\_BEDRRUIMVERMEER[9] = 0.02  
 pct\_BEDRRUIMVERMEER[10] = 0.02  
 pct\_BEDRRUIMVERMEER[11] = 0.02  
 pct\_BEDRRUIMVERMEER[12] = 0.02  
 pct\_BEDRRUIMVERMEER[13] = 0.02  
 pct\_BEDRRUIMVERMEER[14] = 0.03  
 pct\_BEDRRUIMVERMEER[15] = 0.02  
 pct\_BEDRRUIMVERMEER[16] = 0.02  
 pct\_BEDRRUIMVERMEER[17] = 0.03  
 pct\_BEDRRUIMVERMEER[18] = 0.02  
 pct\_BEDRRUIMVERMEER[19] = 0.02  
 pct\_BEDRRUIMVERMEER[20] = 0.02  
 pct\_BEDRRUIMVERMEER[21] = 0.02  
 pct\_BEDRRUIMVERMEER[22] = 0.02  
 pct\_BEDRRUIMVERMEER[23] = 0.02  
 pct\_BEDRRUIMVERMEER[24] = 0.02  
 pct\_BEDRRUIMVERMEER[25] = 0.02  
 pct\_BEDRRUIMVERMEER[26] = 0.002  
 pct\_BEDRRUIMVERMEER[27] = 0.02  
 pct\_BEDRRUIMVERMEER[28] = 0.02  
 pct\_BEDRRUIMVERMEER[29] = 0.02  
 pct\_BEDRRUIMVERMEER[30] = 0.02  
 pct\_BEDRRUIMVERMEER[31] = 0.015  
 pct\_BEDRRUIMVERMEER[32] = 0.02  
 pct\_BEDRRUIMVERMEER[33] = 0.02  
 pct\_BEDRRUIMVERMEER[34] = 0.02  
 pct\_BEDRRUIMVERMEER[35] = 0.02  
 pct\_BEDRRUIMVERMEER[36] = 0.02  
 pct\_BEDRRUIMVERMEER[37] = 0.02  
 pct\_BEDRRUIMVERMEER[38] = 0.02  
 pct\_BEDRRUIMVERMEER[39] = 0.02  
 pct\_BEDRRUIMVERMEER[40] = 0.02  
 pct\_BEDRRUIMVERMIN[van] = 0.002  
 BRUIMBVB[van] = GRAPH(BVBR[van])  
 (0.00, 0.11), (0.2, 0.14), (0.4, 0.18), (0.6, 0.25), (0.8, 0.41), (1.00, 1.00), (1.20, 1.66), (1.40, 1.83),  
 (1.60, 1.92), (1.80, 1.98), (2.00, 2.00)  
 BRUIMBVSM[van] = GRAPH(BVBR[van])  
 (0.00, 1.25), (0.2, 1.24), (0.4, 1.21), (0.6, 1.17), (0.8, 1.10), (1.00, 1.00), (1.20, 0.9), (1.40, 0.83),  
 (1.60, 0.78), (1.80, 0.75), (2.00, 0.74)  
 BEDRIJFSVESTIGINGEN[1](t) = BEDRIJFSVESTIGINGEN[1](t - dt) + (Immigratie\_BV[1] +  
 Binnenlandse\_inmigratie\_BV[1] + Oprichting\_vanuit\_BV[1] + Starters[1] - Emigratie\_BV[1] -  
 Opheffing\_BV[1] - Binnenlandse\_uitmigratie\_BV[1]) \* dt  
 INIT BEDRIJFSVESTIGINGEN[1] = 5663  
 BEDRIJFSVESTIGINGEN[2](t) = BEDRIJFSVESTIGINGEN[2](t - dt) + (Immigratie\_BV[2] +  
 Binnenlandse\_inmigratie\_BV[2] + Oprichting\_vanuit\_BV[2] + Starters[2] - Emigratie\_BV[2] -  
 Opheffing\_BV[2] - Binnenlandse\_uitmigratie\_BV[2]) \* dt  
 INIT BEDRIJFSVESTIGINGEN[2] = 1817  
 BEDRIJFSVESTIGINGEN[3](t) = BEDRIJFSVESTIGINGEN[3](t - dt) + (Immigratie\_BV[3] +  
 Binnenlandse\_inmigratie\_BV[3] + Oprichting\_vanuit\_BV[3] + Starters[3] - Emigratie\_BV[3] -  
 Opheffing\_BV[3] - Binnenlandse\_uitmigratie\_BV[3]) \* dt  
 INIT BEDRIJFSVESTIGINGEN[3] = 12042

$BEDRIJFSVESTIGINGEN[4](t) = BEDRIJFSVESTIGINGEN[4](t - dt) + (Immigratie\_BV[4] + Binnenlandse\_inmigratie\_BV[4] + Oprichting\_vanuit\_BV[4] + Starters[4] - Emigratie\_BV[4] - Opheffing\_BV[4] - Binnenlandse\_uitmigratie\_BV[4]) * dt$   
 INIT  $BEDRIJFSVESTIGINGEN[4] = 9838$   
 $BEDRIJFSVESTIGINGEN[5](t) = BEDRIJFSVESTIGINGEN[5](t - dt) + (Immigratie\_BV[5] + Binnenlandse\_inmigratie\_BV[5] + Oprichting\_vanuit\_BV[5] + Starters[5] - Emigratie\_BV[5] - Opheffing\_BV[5] - Binnenlandse\_uitmigratie\_BV[5]) * dt$   
 INIT  $BEDRIJFSVESTIGINGEN[5] = 4080$   
 $BEDRIJFSVESTIGINGEN[6](t) = BEDRIJFSVESTIGINGEN[6](t - dt) + (Immigratie\_BV[6] + Binnenlandse\_inmigratie\_BV[6] + Oprichting\_vanuit\_BV[6] + Starters[6] - Emigratie\_BV[6] - Opheffing\_BV[6] - Binnenlandse\_uitmigratie\_BV[6]) * dt$   
 INIT  $BEDRIJFSVESTIGINGEN[6] = 4938$   
 $BEDRIJFSVESTIGINGEN[7](t) = BEDRIJFSVESTIGINGEN[7](t - dt) + (Immigratie\_BV[7] + Binnenlandse\_inmigratie\_BV[7] + Oprichting\_vanuit\_BV[7] + Starters[7] - Emigratie\_BV[7] - Opheffing\_BV[7] - Binnenlandse\_uitmigratie\_BV[7]) * dt$   
 INIT  $BEDRIJFSVESTIGINGEN[7] = 3383$   
 $BEDRIJFSVESTIGINGEN[8](t) = BEDRIJFSVESTIGINGEN[8](t - dt) + (Immigratie\_BV[8] + Binnenlandse\_inmigratie\_BV[8] + Oprichting\_vanuit\_BV[8] + Starters[8] - Emigratie\_BV[8] - Opheffing\_BV[8] - Binnenlandse\_uitmigratie\_BV[8]) * dt$   
 INIT  $BEDRIJFSVESTIGINGEN[8] = 3405$   
 $BEDRIJFSVESTIGINGEN[9](t) = BEDRIJFSVESTIGINGEN[9](t - dt) + (Immigratie\_BV[9] + Binnenlandse\_inmigratie\_BV[9] + Oprichting\_vanuit\_BV[9] + Starters[9] - Emigratie\_BV[9] - Opheffing\_BV[9] - Binnenlandse\_uitmigratie\_BV[9]) * dt$   
 INIT  $BEDRIJFSVESTIGINGEN[9] = 3326$   
 $BEDRIJFSVESTIGINGEN[10](t) = BEDRIJFSVESTIGINGEN[10](t - dt) + (Immigratie\_BV[10] + Binnenlandse\_inmigratie\_BV[10] + Oprichting\_vanuit\_BV[10] + Starters[10] - Emigratie\_BV[10] - Opheffing\_BV[10] - Binnenlandse\_uitmigratie\_BV[10]) * dt$   
 INIT  $BEDRIJFSVESTIGINGEN[10] = 9117$   
 $BEDRIJFSVESTIGINGEN[11](t) = BEDRIJFSVESTIGINGEN[11](t - dt) + (Immigratie\_BV[11] + Binnenlandse\_inmigratie\_BV[11] + Oprichting\_vanuit\_BV[11] + Starters[11] - Emigratie\_BV[11] - Opheffing\_BV[11] - Binnenlandse\_uitmigratie\_BV[11]) * dt$   
 INIT  $BEDRIJFSVESTIGINGEN[11] = 3212$   
 $BEDRIJFSVESTIGINGEN[12](t) = BEDRIJFSVESTIGINGEN[12](t - dt) + (Immigratie\_BV[12] + Binnenlandse\_inmigratie\_BV[12] + Oprichting\_vanuit\_BV[12] + Starters[12] - Emigratie\_BV[12] - Opheffing\_BV[12] - Binnenlandse\_uitmigratie\_BV[12]) * dt$   
 INIT  $BEDRIJFSVESTIGINGEN[12] = 14212$   
 $BEDRIJFSVESTIGINGEN[13](t) = BEDRIJFSVESTIGINGEN[13](t - dt) + (Immigratie\_BV[13] + Binnenlandse\_inmigratie\_BV[13] + Oprichting\_vanuit\_BV[13] + Starters[13] - Emigratie\_BV[13] - Opheffing\_BV[13] - Binnenlandse\_uitmigratie\_BV[13]) * dt$   
 INIT  $BEDRIJFSVESTIGINGEN[13] = 12463$   
 $BEDRIJFSVESTIGINGEN[14](t) = BEDRIJFSVESTIGINGEN[14](t - dt) + (Immigratie\_BV[14] + Binnenlandse\_inmigratie\_BV[14] + Oprichting\_vanuit\_BV[14] + Starters[14] - Emigratie\_BV[14] - Opheffing\_BV[14] - Binnenlandse\_uitmigratie\_BV[14]) * dt$   
 INIT  $BEDRIJFSVESTIGINGEN[14] = 9940$   
 $BEDRIJFSVESTIGINGEN[15](t) = BEDRIJFSVESTIGINGEN[15](t - dt) + (Immigratie\_BV[15] + Binnenlandse\_inmigratie\_BV[15] + Oprichting\_vanuit\_BV[15] + Starters[15] - Emigratie\_BV[15] - Opheffing\_BV[15] - Binnenlandse\_uitmigratie\_BV[15]) * dt$   
 INIT  $BEDRIJFSVESTIGINGEN[15] = 15307$   
 $BEDRIJFSVESTIGINGEN[16](t) = BEDRIJFSVESTIGINGEN[16](t - dt) + (Immigratie\_BV[16] + Binnenlandse\_inmigratie\_BV[16] + Oprichting\_vanuit\_BV[16] + Starters[16] - Emigratie\_BV[16] - Opheffing\_BV[16] - Binnenlandse\_uitmigratie\_BV[16]) * dt$   
 INIT  $BEDRIJFSVESTIGINGEN[16] = 5017$   
 $BEDRIJFSVESTIGINGEN[17](t) = BEDRIJFSVESTIGINGEN[17](t - dt) + (Immigratie\_BV[17] + Binnenlandse\_inmigratie\_BV[17] + Oprichting\_vanuit\_BV[17] + Starters[17] - Emigratie\_BV[17] - Opheffing\_BV[17] - Binnenlandse\_uitmigratie\_BV[17]) * dt$



INIT BEDRIJFSVESTIGINGEN[17] = 22837  
 BEDRIJFSVESTIGINGEN[18](t) = BEDRIJFSVESTIGINGEN[18](t - dt) +  
 (Immigratie\_BV[18] + Binnenlandse\_inmigratie\_BV[18] + Oprichting\_vanuit\_BV[18] +  
 Starters[18] - Emigratie\_BV[18] - Opheffing\_BV[18] - Binnenlandse\_uitmigratie\_BV[18]) \* dt  
 INIT BEDRIJFSVESTIGINGEN[18] = 7392  
 BEDRIJFSVESTIGINGEN[19](t) = BEDRIJFSVESTIGINGEN[19](t - dt) +  
 (Immigratie\_BV[19] + Binnenlandse\_inmigratie\_BV[19] + Oprichting\_vanuit\_BV[19] +  
 Starters[19] - Emigratie\_BV[19] - Opheffing\_BV[19] - Binnenlandse\_uitmigratie\_BV[19]) \* dt  
 INIT BEDRIJFSVESTIGINGEN[19] = 4512  
 BEDRIJFSVESTIGINGEN[20](t) = BEDRIJFSVESTIGINGEN[20](t - dt) +  
 (Immigratie\_BV[20] + Binnenlandse\_inmigratie\_BV[20] + Oprichting\_vanuit\_BV[20] +  
 Starters[20] - Emigratie\_BV[20] - Opheffing\_BV[20] - Binnenlandse\_uitmigratie\_BV[20]) \* dt  
 INIT BEDRIJFSVESTIGINGEN[20] = 3763  
 BEDRIJFSVESTIGINGEN[21](t) = BEDRIJFSVESTIGINGEN[21](t - dt) +  
 (Immigratie\_BV[21] + Binnenlandse\_inmigratie\_BV[21] + Oprichting\_vanuit\_BV[21] +  
 Starters[21] - Emigratie\_BV[21] - Opheffing\_BV[21] - Binnenlandse\_uitmigratie\_BV[21]) \* dt  
 INIT BEDRIJFSVESTIGINGEN[21] = 8027  
 BEDRIJFSVESTIGINGEN[22](t) = BEDRIJFSVESTIGINGEN[22](t - dt) +  
 (Immigratie\_BV[22] + Binnenlandse\_inmigratie\_BV[22] + Oprichting\_vanuit\_BV[22] +  
 Starters[22] - Emigratie\_BV[22] - Opheffing\_BV[22] - Binnenlandse\_uitmigratie\_BV[22]) \* dt  
 INIT BEDRIJFSVESTIGINGEN[22] = 3522  
 BEDRIJFSVESTIGINGEN[23](t) = BEDRIJFSVESTIGINGEN[23](t - dt) +  
 (Immigratie\_BV[23] + Binnenlandse\_inmigratie\_BV[23] + Oprichting\_vanuit\_BV[23] +  
 Starters[23] - Emigratie\_BV[23] - Opheffing\_BV[23] - Binnenlandse\_uitmigratie\_BV[23]) \* dt  
 INIT BEDRIJFSVESTIGINGEN[23] = 42252  
 BEDRIJFSVESTIGINGEN[24](t) = BEDRIJFSVESTIGINGEN[24](t - dt) +  
 (Immigratie\_BV[24] + Binnenlandse\_inmigratie\_BV[24] + Oprichting\_vanuit\_BV[24] +  
 Starters[24] - Emigratie\_BV[24] - Opheffing\_BV[24] - Binnenlandse\_uitmigratie\_BV[24]) \* dt  
 INIT BEDRIJFSVESTIGINGEN[24] = 8333  
 BEDRIJFSVESTIGINGEN[25](t) = BEDRIJFSVESTIGINGEN[25](t - dt) +  
 (Immigratie\_BV[25] + Binnenlandse\_inmigratie\_BV[25] + Oprichting\_vanuit\_BV[25] +  
 Starters[25] - Emigratie\_BV[25] - Opheffing\_BV[25] - Binnenlandse\_uitmigratie\_BV[25]) \* dt  
 INIT BEDRIJFSVESTIGINGEN[25] = 8540  
 BEDRIJFSVESTIGINGEN[26](t) = BEDRIJFSVESTIGINGEN[26](t - dt) +  
 (Immigratie\_BV[26] + Binnenlandse\_inmigratie\_BV[26] + Oprichting\_vanuit\_BV[26] +  
 Starters[26] - Emigratie\_BV[26] - Opheffing\_BV[26] - Binnenlandse\_uitmigratie\_BV[26]) \* dt  
 INIT BEDRIJFSVESTIGINGEN[26] = 25103  
 BEDRIJFSVESTIGINGEN[27](t) = BEDRIJFSVESTIGINGEN[27](t - dt) +  
 (Immigratie\_BV[27] + Binnenlandse\_inmigratie\_BV[27] + Oprichting\_vanuit\_BV[27] +  
 Starters[27] - Emigratie\_BV[27] - Opheffing\_BV[27] - Binnenlandse\_uitmigratie\_BV[27]) \* dt  
 INIT BEDRIJFSVESTIGINGEN[27] = 4507  
 BEDRIJFSVESTIGINGEN[28](t) = BEDRIJFSVESTIGINGEN[28](t - dt) +  
 (Immigratie\_BV[28] + Binnenlandse\_inmigratie\_BV[28] + Oprichting\_vanuit\_BV[28] +  
 Starters[28] - Emigratie\_BV[28] - Opheffing\_BV[28] - Binnenlandse\_uitmigratie\_BV[28]) \* dt  
 INIT BEDRIJFSVESTIGINGEN[28] = 7396  
 BEDRIJFSVESTIGINGEN[29](t) = BEDRIJFSVESTIGINGEN[29](t - dt) +  
 (Immigratie\_BV[29] + Binnenlandse\_inmigratie\_BV[29] + Oprichting\_vanuit\_BV[29] +  
 Starters[29] - Emigratie\_BV[29] - Opheffing\_BV[29] - Binnenlandse\_uitmigratie\_BV[29]) \* dt  
 INIT BEDRIJFSVESTIGINGEN[29] = 37627  
 BEDRIJFSVESTIGINGEN[30](t) = BEDRIJFSVESTIGINGEN[30](t - dt) +  
 (Immigratie\_BV[30] + Binnenlandse\_inmigratie\_BV[30] + Oprichting\_vanuit\_BV[30] +  
 Starters[30] - Emigratie\_BV[30] - Opheffing\_BV[30] - Binnenlandse\_uitmigratie\_BV[30]) \* dt  
 INIT BEDRIJFSVESTIGINGEN[30] = 8943

$BEDRIJFSVESTIGINGEN[31](t) = BEDRIJFSVESTIGINGEN[31](t - dt) +$   
 $(Immigratie\_BV[31] + Binnenlandse\_inmigratie\_BV[31] + Oprichting\_vanuit\_BV[31] +$   
 $Starters[31] - Emigratie\_BV[31] - Opheffing\_BV[31] - Binnenlandse\_uitmigratie\_BV[31]) * dt$   
 $INIT\ BEDRIJFSVESTIGINGEN[31] = 4755$

$BEDRIJFSVESTIGINGEN[32](t) = BEDRIJFSVESTIGINGEN[32](t - dt) +$   
 $(Immigratie\_BV[32] + Binnenlandse\_inmigratie\_BV[32] + Oprichting\_vanuit\_BV[32] +$   
 $Starters[32] - Emigratie\_BV[32] - Opheffing\_BV[32] - Binnenlandse\_uitmigratie\_BV[32]) * dt$   
 $INIT\ BEDRIJFSVESTIGINGEN[32] = 8117$

$BEDRIJFSVESTIGINGEN[33](t) = BEDRIJFSVESTIGINGEN[33](t - dt) +$   
 $(Immigratie\_BV[33] + Binnenlandse\_inmigratie\_BV[33] + Oprichting\_vanuit\_BV[33] +$   
 $Starters[33] - Emigratie\_BV[33] - Opheffing\_BV[33] - Binnenlandse\_uitmigratie\_BV[33]) * dt$   
 $INIT\ BEDRIJFSVESTIGINGEN[33] = 13438$

$BEDRIJFSVESTIGINGEN[34](t) = BEDRIJFSVESTIGINGEN[34](t - dt) +$   
 $(Immigratie\_BV[34] + Binnenlandse\_inmigratie\_BV[34] + Oprichting\_vanuit\_BV[34] +$   
 $Starters[34] - Emigratie\_BV[34] - Opheffing\_BV[34] - Binnenlandse\_uitmigratie\_BV[34]) * dt$   
 $INIT\ BEDRIJFSVESTIGINGEN[34] = 10744$

$BEDRIJFSVESTIGINGEN[35](t) = BEDRIJFSVESTIGINGEN[35](t - dt) +$   
 $(Immigratie\_BV[35] + Binnenlandse\_inmigratie\_BV[35] + Oprichting\_vanuit\_BV[35] +$   
 $Starters[35] - Emigratie\_BV[35] - Opheffing\_BV[35] - Binnenlandse\_uitmigratie\_BV[35]) * dt$   
 $INIT\ BEDRIJFSVESTIGINGEN[35] = 10401$

$BEDRIJFSVESTIGINGEN[36](t) = BEDRIJFSVESTIGINGEN[36](t - dt) +$   
 $(Immigratie\_BV[36] + Binnenlandse\_inmigratie\_BV[36] + Oprichting\_vanuit\_BV[36] +$   
 $Starters[36] - Emigratie\_BV[36] - Opheffing\_BV[36] - Binnenlandse\_uitmigratie\_BV[36]) * dt$   
 $INIT\ BEDRIJFSVESTIGINGEN[36] = 12335$

$BEDRIJFSVESTIGINGEN[37](t) = BEDRIJFSVESTIGINGEN[37](t - dt) +$   
 $(Immigratie\_BV[37] + Binnenlandse\_inmigratie\_BV[37] + Oprichting\_vanuit\_BV[37] +$   
 $Starters[37] - Emigratie\_BV[37] - Opheffing\_BV[37] - Binnenlandse\_uitmigratie\_BV[37]) * dt$   
 $INIT\ BEDRIJFSVESTIGINGEN[37] = 6166$

$BEDRIJFSVESTIGINGEN[38](t) = BEDRIJFSVESTIGINGEN[38](t - dt) +$   
 $(Immigratie\_BV[38] + Binnenlandse\_inmigratie\_BV[38] + Oprichting\_vanuit\_BV[38] +$   
 $Starters[38] - Emigratie\_BV[38] - Opheffing\_BV[38] - Binnenlandse\_uitmigratie\_BV[38]) * dt$   
 $INIT\ BEDRIJFSVESTIGINGEN[38] = 5793$

$BEDRIJFSVESTIGINGEN[39](t) = BEDRIJFSVESTIGINGEN[39](t - dt) +$   
 $(Immigratie\_BV[39] + Binnenlandse\_inmigratie\_BV[39] + Oprichting\_vanuit\_BV[39] +$   
 $Starters[39] - Emigratie\_BV[39] - Opheffing\_BV[39] - Binnenlandse\_uitmigratie\_BV[39]) * dt$   
 $INIT\ BEDRIJFSVESTIGINGEN[39] = 17320$

$BEDRIJFSVESTIGINGEN[40](t) = BEDRIJFSVESTIGINGEN[40](t - dt) +$   
 $(Immigratie\_BV[40] + Binnenlandse\_inmigratie\_BV[40] + Oprichting\_vanuit\_BV[40] +$   
 $Starters[40] - Emigratie\_BV[40] - Opheffing\_BV[40] - Binnenlandse\_uitmigratie\_BV[40]) * dt$   
 $INIT\ BEDRIJFSVESTIGINGEN[40] = 4000$

$Immigratie\_BV[van] = BEDRIJFSVESTIGINGEN[van]*pct\_IMM\_BV[van]*IMMF\_BV[van]$   
 $Binnenlandse\_inmigratie\_BV[van] = INM\_BV[van]$   
 $Oprichting\_vanuit\_BV[van] =$   
 $(BEDRIJFSVESTIGINGEN[van]*pct\_OPR\_bestaande\_BV[van])*AOPRBESTBVM[van]$   
 $Starters[van] = beroepsbevolking\_wonend[van]*pct\_OPR\_starters[van]*ASTM[van]$   
 $Emigratie\_BV[van] = BEDRIJFSVESTIGINGEN[van]*pct\_EM\_BV[van]*EMF\_BV[van]$   
 $Opheffing\_BV[van] = BEDRIJFSVESTIGINGEN[van]*pct\_OPH\_BV[van]*AOPHM[van]$   
 $Binnenlandse\_uitmigratie\_BV[van] = UITM\_BV[van]$   
 $EMF\_BV[van] = ABBM\_EM\_BV[van]*ABRUIMM\_EM\_BV[van]$   
 $IMMF\_BV[van] = ABBM\_IMM\_BV[van]*ABRUIMM\_IMM\_BV[van]$   
 $INMF\_BV[van] = ABBM\_INM\_BV[van]*ABRUIMM\_INM\_BV[van]$   
 $pct\_OPH\_BV[1] = 0.04$   
 $pct\_OPH\_BV[2] = 0.04$   
 $pct\_OPH\_BV[3] = 0.04$   
 $pct\_OPH\_BV[4] = 0.035$

pct\_OPH\_BV[5] = 0.035  
pct\_OPH\_BV[6] = 0.035  
pct\_OPH\_BV[7] = 0.03  
pct\_OPH\_BV[8] = 0.035  
pct\_OPH\_BV[9] = 0.025  
pct\_OPH\_BV[10] = 0.035  
pct\_OPH\_BV[11] = 0.054  
pct\_OPH\_BV[12] = 0.04  
pct\_OPH\_BV[13] = 0.035  
pct\_OPH\_BV[14] = 0.035  
pct\_OPH\_BV[15] = 0.04  
pct\_OPH\_BV[16] = 0.035  
pct\_OPH\_BV[17] = 0.04  
pct\_OPH\_BV[18] = 0.025  
pct\_OPH\_BV[19] = 0.03  
pct\_OPH\_BV[20] = 0.045  
pct\_OPH\_BV[21] = 0.064  
pct\_OPH\_BV[22] = 0.05  
pct\_OPH\_BV[23] = 0.055  
pct\_OPH\_BV[24] = 0.045  
pct\_OPH\_BV[25] = 0.035  
pct\_OPH\_BV[26] = 0.05  
pct\_OPH\_BV[27] = 0.04  
pct\_OPH\_BV[28] = 0.035  
pct\_OPH\_BV[29] = 0.04  
pct\_OPH\_BV[30] = 0.035  
pct\_OPH\_BV[31] = 0.03  
pct\_OPH\_BV[32] = 0.027  
pct\_OPH\_BV[33] = 0.05  
pct\_OPH\_BV[34] = 0.04  
pct\_OPH\_BV[35] = 0.035  
pct\_OPH\_BV[36] = 0.03  
pct\_OPH\_BV[37] = 0.035  
pct\_OPH\_BV[38] = 0.035  
pct\_OPH\_BV[39] = 0.045  
pct\_OPH\_BV[40] = 0.03  
pct\_OPR\_bestaande\_BV[1] = 0.03  
pct\_OPR\_bestaande\_BV[2] = 0.015  
pct\_OPR\_bestaande\_BV[3] = 0.026  
pct\_OPR\_bestaande\_BV[4] = 0.028  
pct\_OPR\_bestaande\_BV[5] = 0.025  
pct\_OPR\_bestaande\_BV[6] = 0.025  
pct\_OPR\_bestaande\_BV[7] = 0.04  
pct\_OPR\_bestaande\_BV[8] = 0.04  
pct\_OPR\_bestaande\_BV[9] = 0.04  
pct\_OPR\_bestaande\_BV[10] = 0.04  
pct\_OPR\_bestaande\_BV[11] = 0.05  
pct\_OPR\_bestaande\_BV[12] = 0.045  
pct\_OPR\_bestaande\_BV[13] = 0.04  
pct\_OPR\_bestaande\_BV[14] = 0.045  
pct\_OPR\_bestaande\_BV[15] = 0.03  
pct\_OPR\_bestaande\_BV[16] = 0.035  
pct\_OPR\_bestaande\_BV[17] = 0.036  
pct\_OPR\_bestaande\_BV[18] = 0.04  
pct\_OPR\_bestaande\_BV[19] = 0.045

pct\_OPR\_bestaande\_BV[20] = 0.025  
pct\_OPR\_bestaande\_BV[21] = 0.04  
pct\_OPR\_bestaande\_BV[22] = 0.05  
pct\_OPR\_bestaande\_BV[23] = 0.04  
pct\_OPR\_bestaande\_BV[24] = 0.026  
pct\_OPR\_bestaande\_BV[25] = 0.035  
pct\_OPR\_bestaande\_BV[26] = 0.035  
pct\_OPR\_bestaande\_BV[27] = 0.04  
pct\_OPR\_bestaande\_BV[28] = 0.04  
pct\_OPR\_bestaande\_BV[29] = 0.03  
pct\_OPR\_bestaande\_BV[30] = 0.035  
pct\_OPR\_bestaande\_BV[31] = 0.026  
pct\_OPR\_bestaande\_BV[32] = 0.03  
pct\_OPR\_bestaande\_BV[33] = 0.045  
pct\_OPR\_bestaande\_BV[34] = 0.04  
pct\_OPR\_bestaande\_BV[35] = 0.04  
pct\_OPR\_bestaande\_BV[36] = 0.04  
pct\_OPR\_bestaande\_BV[37] = 0.04  
pct\_OPR\_bestaande\_BV[38] = 0.034  
pct\_OPR\_bestaande\_BV[39] = 0.035  
pct\_OPR\_bestaande\_BV[40] = 0.055  
pct\_OPR\_ST[1] = 0.002  
pct\_OPR\_ST[2] = 0.001  
pct\_OPR\_ST[3] = 0.002  
pct\_OPR\_ST[4] = 0.002  
pct\_OPR\_ST[5] = 0.002  
pct\_OPR\_ST[6] = 0.002  
pct\_OPR\_ST[7] = 0.002  
pct\_OPR\_ST[8] = 0.002  
pct\_OPR\_ST[9] = 0.002  
pct\_OPR\_ST[10] = 0.002  
pct\_OPR\_ST[11] = 0.002  
pct\_OPR\_ST[12] = 0.002  
pct\_OPR\_ST[13] = 0.003  
pct\_OPR\_ST[14] = 0.002  
pct\_OPR\_ST[15] = 0.002  
pct\_OPR\_ST[16] = 0.004  
pct\_OPR\_ST[17] = 0.003  
pct\_OPR\_ST[18] = 0.002  
pct\_OPR\_ST[19] = 0.002  
pct\_OPR\_ST[20] = 0.002  
pct\_OPR\_ST[21] = 0.002  
pct\_OPR\_ST[22] = 0.0001  
pct\_OPR\_ST[23] = 0.003  
pct\_OPR\_ST[24] = 0.001  
pct\_OPR\_ST[25] = 0.002  
pct\_OPR\_ST[26] = 0.001  
pct\_OPR\_ST[27] = 0.0035  
pct\_OPR\_ST[28] = 0.002  
pct\_OPR\_ST[29] = 0.002  
pct\_OPR\_ST[30] = 0.002  
pct\_OPR\_ST[31] = 0.002  
pct\_OPR\_ST[32] = 0.0025  
pct\_OPR\_ST[33] = 0.003  
pct\_OPR\_ST[34] = 0.002

pct\_OPR\_ST[35] = 0.003  
 pct\_OPR\_ST[36] = 0.002  
 pct\_OPR\_ST[37] = 0.002  
 pct\_OPR\_ST[38] = 0.002  
 pct\_OPR\_ST[39] = 0.002  
 pct\_OPR\_ST[40] = 0.004  
 pct\_OPR\_starters[van] = pct\_OPR\_ST[van]  
 pct\_UITM\_BV[van] = 0.02  
 pot\_INM\_BV[van] = BEDRIJFSVESTIGINGEN[van]\*INMF\_BV[van]  
 pot\_UITM\_BV[van] = BEDRIJFSVESTIGINGEN[van]\*pct\_UITM\_BV[van]\*UITMF\_BV[van]  
 UITMF\_BV[van] = ABBM\_UITM\_BV[van]\*ABRUIMM\_UITM\_BV[van]  
 ABRUIMM\_EM\_BV[van] = GRAPH(BVBR[van])  
 (0.00, 0.79), (0.2, 0.81), (0.4, 0.83), (0.6, 0.87), (0.8, 0.92), (1.00, 1.00), (1.20, 1.07), (1.40, 1.12),  
 (1.60, 1.16), (1.80, 1.18), (2.00, 1.19)  
 ABRUIMM\_IMM\_BV[van] = GRAPH(BVBR[van])  
 (0.00, 1.20), (0.2, 1.20), (0.4, 1.18), (0.6, 1.14), (0.8, 1.07), (1.00, 1.00), (1.20, 0.93), (1.40, 0.88),  
 (1.60, 0.83), (1.80, 0.81), (2.00, 0.79)  
 ABRUIMM\_INM\_BV[van] = GRAPH(BVBR[van])  
 (0.00, 1.67), (0.2, 1.67), (0.4, 1.62), (0.6, 1.48), (0.8, 1.26), (1.00, 1.00), (1.20, 0.75), (1.40, 0.51),  
 (1.60, 0.36), (1.80, 0.32), (2.00, 0.32)  
 ABRUIMM\_UITM\_BV[van] = GRAPH(BVBR[van])  
 (0.00, 0.57), (0.2, 0.6), (0.4, 0.64), (0.6, 0.71), (0.8, 0.81), (1.00, 1.00), (1.20, 1.20), (1.40, 1.33),  
 (1.60, 1.38), (1.80, 1.42), (2.00, 1.43)  
 AOPHM[van] = GRAPH(BBAR[van])  
 (0.00, 1.09), (0.2, 1.08), (0.4, 1.08), (0.6, 1.06), (0.8, 1.04), (1.00, 1.00), (1.20, 0.95), (1.40, 0.93),  
 (1.60, 0.91), (1.80, 0.9), (2.00, 0.89)  
 AOPRBESTBVM[van] = GRAPH(BBAR[van])  
 (0.00, 0.92), (0.2, 0.92), (0.4, 0.94), (0.6, 0.95), (0.8, 0.97), (1.00, 1.00), (1.20, 1.02), (1.40, 1.04),  
 (1.60, 1.06), (1.80, 1.07), (2.00, 1.07)  
 ASTM[van] = GRAPH(BBAR[van])  
 (0.00, 0.9), (0.2, 0.9), (0.4, 0.91), (0.6, 0.93), (0.8, 0.95), (1.00, 1.00), (1.20, 1.04), (1.40, 1.05),  
 (1.60, 1.06), (1.80, 1.07), (2.00, 1.07)  
 BEVOLKINfG[1](t) = BEVOLKINfG[1](t - dt) + (Immigratie\_BEV[1] +  
 Binnenlandse\_inmigratie\_BEV[1] + Geboorte\_BEV[1] - Emigratie\_BEV[1] - Sterfte\_BEV[1] -  
 Binnenlandse\_uitmigratie\_BEV[1]) \* dt  
 INIT BEVOLKINfG[1] = 144500  
 BEVOLKINfG[2](t) = BEVOLKINfG[2](t - dt) + (Immigratie\_BEV[2] +  
 Binnenlandse\_inmigratie\_BEV[2] + Geboorte\_BEV[2] - Emigratie\_BEV[2] - Sterfte\_BEV[2] -  
 Binnenlandse\_uitmigratie\_BEV[2]) \* dt  
 INIT BEVOLKINfG[2] = 52500  
 BEVOLKINfG[3](t) = BEVOLKINfG[3](t - dt) + (Immigratie\_BEV[3] +  
 Binnenlandse\_inmigratie\_BEV[3] + Geboorte\_BEV[3] - Emigratie\_BEV[3] - Sterfte\_BEV[3] -  
 Binnenlandse\_uitmigratie\_BEV[3]) \* dt  
 INIT BEVOLKINfG[3] = 326375  
 BEVOLKINfG[4](t) = BEVOLKINfG[4](t - dt) + (Immigratie\_BEV[4] +  
 Binnenlandse\_inmigratie\_BEV[4] + Geboorte\_BEV[4] - Emigratie\_BEV[4] - Sterfte\_BEV[4] -  
 Binnenlandse\_uitmigratie\_BEV[4]) \* dt  
 INIT BEVOLKINfG[4] = 285300  
 BEVOLKINfG[5](t) = BEVOLKINfG[5](t - dt) + (Immigratie\_BEV[5] +  
 Binnenlandse\_inmigratie\_BEV[5] + Geboorte\_BEV[5] - Emigratie\_BEV[5] - Sterfte\_BEV[5] -  
 Binnenlandse\_uitmigratie\_BEV[5]) \* dt  
 INIT BEVOLKINfG[5] = 87400  
 BEVOLKINfG[6](t) = BEVOLKINfG[6](t - dt) + (Immigratie\_BEV[6] +  
 Binnenlandse\_inmigratie\_BEV[6] + Geboorte\_BEV[6] - Emigratie\_BEV[6] - Sterfte\_BEV[6] -  
 Binnenlandse\_uitmigratie\_BEV[6]) \* dt

INIT BEVOLKINfG[6] = 151000  
 BEVOLKINfG[7](t) = BEVOLKINfG[7](t - dt) + (Immigratie\_BEV[7] +  
 Binnenlandse\_inmigratie\_BEV[7] + Geboorte\_BEV[7] - Emigratie\_BEV[7] - Sterfte\_BEV[7] -  
 Binnenlandse\_uitmigratie\_BEV[7]) \* dt  
 INIT BEVOLKINfG[7] = 128800  
 BEVOLKINfG[8](t) = BEVOLKINfG[8](t - dt) + (Immigratie\_BEV[8] +  
 Binnenlandse\_inmigratie\_BEV[8] + Geboorte\_BEV[8] - Emigratie\_BEV[8] - Sterfte\_BEV[8] -  
 Binnenlandse\_uitmigratie\_BEV[8]) \* dt  
 INIT BEVOLKINfG[8] = 141400  
 BEVOLKINfG[9](t) = BEVOLKINfG[9](t - dt) + (Immigratie\_BEV[9] +  
 Binnenlandse\_inmigratie\_BEV[9] + Geboorte\_BEV[9] - Emigratie\_BEV[9] - Sterfte\_BEV[9] -  
 Binnenlandse\_uitmigratie\_BEV[9]) \* dt  
 INIT BEVOLKINfG[9] = 98420  
 BEVOLKINfG[10](t) = BEVOLKINfG[10](t - dt) + (Immigratie\_BEV[10] +  
 Binnenlandse\_inmigratie\_BEV[10] + Geboorte\_BEV[10] - Emigratie\_BEV[10] -  
 Sterfte\_BEV[10] - Binnenlandse\_uitmigratie\_BEV[10]) \* dt  
 INIT BEVOLKINfG[10] = 266400  
 BEVOLKINfG[11](t) = BEVOLKINfG[11](t - dt) + (Immigratie\_BEV[11] +  
 Binnenlandse\_inmigratie\_BEV[11] + Geboorte\_BEV[11] - Emigratie\_BEV[11] -  
 Sterfte\_BEV[11] - Binnenlandse\_uitmigratie\_BEV[11]) \* dt  
 INIT BEVOLKINfG[11] = 117000  
 BEVOLKINfG[12](t) = BEVOLKINfG[12](t - dt) + (Immigratie\_BEV[12] +  
 Binnenlandse\_inmigratie\_BEV[12] + Geboorte\_BEV[12] - Emigratie\_BEV[12] -  
 Sterfte\_BEV[12] - Binnenlandse\_uitmigratie\_BEV[12]) \* dt  
 INIT BEVOLKINfG[12] = 504725  
 BEVOLKINfG[13](t) = BEVOLKINfG[13](t - dt) + (Immigratie\_BEV[13] +  
 Binnenlandse\_inmigratie\_BEV[13] + Geboorte\_BEV[13] - Emigratie\_BEV[13] -  
 Sterfte\_BEV[13] - Binnenlandse\_uitmigratie\_BEV[13]) \* dt  
 INIT BEVOLKINfG[13] = 470200  
 BEVOLKINfG[14](t) = BEVOLKINfG[14](t - dt) + (Immigratie\_BEV[14] +  
 Binnenlandse\_inmigratie\_BEV[14] + Geboorte\_BEV[14] - Emigratie\_BEV[14] -  
 Sterfte\_BEV[14] - Binnenlandse\_uitmigratie\_BEV[14]) \* dt  
 INIT BEVOLKINfG[14] = 316700  
 BEVOLKINfG[15](t) = BEVOLKINfG[15](t - dt) + (Immigratie\_BEV[15] +  
 Binnenlandse\_inmigratie\_BEV[15] + Geboorte\_BEV[15] - Emigratie\_BEV[15] -  
 Sterfte\_BEV[15] - Binnenlandse\_uitmigratie\_BEV[15]) \* dt  
 INIT BEVOLKINfG[15] = 580000  
 BEVOLKINfG[16](t) = BEVOLKINfG[16](t - dt) + (Immigratie\_BEV[16] +  
 Binnenlandse\_inmigratie\_BEV[16] + Geboorte\_BEV[16] - Emigratie\_BEV[16] -  
 Sterfte\_BEV[16] - Binnenlandse\_uitmigratie\_BEV[16]) \* dt  
 INIT BEVOLKINfG[16] = 153000  
 BEVOLKINfG[17](t) = BEVOLKINfG[17](t - dt) + (Immigratie\_BEV[17] +  
 Binnenlandse\_inmigratie\_BEV[17] + Geboorte\_BEV[17] - Emigratie\_BEV[17] -  
 Sterfte\_BEV[17] - Binnenlandse\_uitmigratie\_BEV[17]) \* dt  
 INIT BEVOLKINfG[17] = 835200  
 BEVOLKINfG[18](t) = BEVOLKINfG[18](t - dt) + (Immigratie\_BEV[18] +  
 Binnenlandse\_inmigratie\_BEV[18] + Geboorte\_BEV[18] - Emigratie\_BEV[18] -  
 Sterfte\_BEV[18] - Binnenlandse\_uitmigratie\_BEV[18]) \* dt  
 INIT BEVOLKINfG[18] = 230750  
 BEVOLKINfG[19](t) = BEVOLKINfG[19](t - dt) + (Immigratie\_BEV[19] +  
 Binnenlandse\_inmigratie\_BEV[19] + Geboorte\_BEV[19] - Emigratie\_BEV[19] -  
 Sterfte\_BEV[19] - Binnenlandse\_uitmigratie\_BEV[19]) \* dt  
 INIT BEVOLKINfG[19] = 147000

$BEVOLKINfG[20](t) = BEVOLKINfG[20](t - dt) + (Immigratie\_BEV[20] +$   
 $Binnenlandse\_inmigratie\_BEV[20] + Geboorte\_BEV[20] - Emigratie\_BEV[20] -$   
 $Sterfte\_BEV[20] - Binnenlandse\_uitmigratie\_BEV[20]) * dt$   
 INIT BEVOLKINfG[20] = 164000  
 $BEVOLKINfG[21](t) = BEVOLKINfG[21](t - dt) + (Immigratie\_BEV[21] +$   
 $Binnenlandse\_inmigratie\_BEV[21] + Geboorte\_BEV[21] - Emigratie\_BEV[21] -$   
 $Sterfte\_BEV[21] - Binnenlandse\_uitmigratie\_BEV[21]) * dt$   
 INIT BEVOLKINfG[21] = 244475  
 $BEVOLKINfG[22](t) = BEVOLKINfG[22](t - dt) + (Immigratie\_BEV[22] +$   
 $Binnenlandse\_inmigratie\_BEV[22] + Geboorte\_BEV[22] - Emigratie\_BEV[22] -$   
 $Sterfte\_BEV[22] - Binnenlandse\_uitmigratie\_BEV[22]) * dt$   
 INIT BEVOLKINfG[22] = 129000  
 $BEVOLKINfG[23](t) = BEVOLKINfG[23](t - dt) + (Immigratie\_BEV[23] +$   
 $Binnenlandse\_inmigratie\_BEV[23] + Geboorte\_BEV[23] - Emigratie\_BEV[23] -$   
 $Sterfte\_BEV[23] - Binnenlandse\_uitmigratie\_BEV[23]) * dt$   
 INIT BEVOLKINfG[23] = 1097000  
 $BEVOLKINfG[24](t) = BEVOLKINfG[24](t - dt) + (Immigratie\_BEV[24] +$   
 $Binnenlandse\_inmigratie\_BEV[24] + Geboorte\_BEV[24] - Emigratie\_BEV[24] -$   
 $Sterfte\_BEV[24] - Binnenlandse\_uitmigratie\_BEV[24]) * dt$   
 INIT BEVOLKINfG[24] = 235000  
 $BEVOLKINfG[25](t) = BEVOLKINfG[25](t - dt) + (Immigratie\_BEV[25] +$   
 $Binnenlandse\_inmigratie\_BEV[25] + Geboorte\_BEV[25] - Emigratie\_BEV[25] -$   
 $Sterfte\_BEV[25] - Binnenlandse\_uitmigratie\_BEV[25]) * dt$   
 INIT BEVOLKINfG[25] = 315400  
 $BEVOLKINfG[26](t) = BEVOLKINfG[26](t - dt) + (Immigratie\_BEV[26] +$   
 $Binnenlandse\_inmigratie\_BEV[26] + Geboorte\_BEV[26] - Emigratie\_BEV[26] -$   
 $Sterfte\_BEV[26] - Binnenlandse\_uitmigratie\_BEV[26]) * dt$   
 INIT BEVOLKINfG[26] = 724000  
 $BEVOLKINfG[27](t) = BEVOLKINfG[27](t - dt) + (Immigratie\_BEV[27] +$   
 $Binnenlandse\_inmigratie\_BEV[27] + Geboorte\_BEV[27] - Emigratie\_BEV[27] -$   
 $Sterfte\_BEV[27] - Binnenlandse\_uitmigratie\_BEV[27]) * dt$   
 INIT BEVOLKINfG[27] = 183000  
 $BEVOLKINfG[28](t) = BEVOLKINfG[28](t - dt) + (Immigratie\_BEV[28] +$   
 $Binnenlandse\_inmigratie\_BEV[28] + Geboorte\_BEV[28] - Emigratie\_BEV[28] -$   
 $Sterfte\_BEV[28] - Binnenlandse\_uitmigratie\_BEV[28]) * dt$   
 INIT BEVOLKINfG[28] = 220550  
 $BEVOLKINfG[29](t) = BEVOLKINfG[29](t - dt) + (Immigratie\_BEV[29] +$   
 $Binnenlandse\_inmigratie\_BEV[29] + Geboorte\_BEV[29] - Emigratie\_BEV[29] -$   
 $Sterfte\_BEV[29] - Binnenlandse\_uitmigratie\_BEV[29]) * dt$   
 INIT BEVOLKINfG[29] = 1223500  
 $BEVOLKINfG[30](t) = BEVOLKINfG[30](t - dt) + (Immigratie\_BEV[30] +$   
 $Binnenlandse\_inmigratie\_BEV[30] + Geboorte\_BEV[30] - Emigratie\_BEV[30] -$   
 $Sterfte\_BEV[30] - Binnenlandse\_uitmigratie\_BEV[30]) * dt$   
 INIT BEVOLKINfG[30] = 320800  
 $BEVOLKINfG[31](t) = BEVOLKINfG[31](t - dt) + (Immigratie\_BEV[31] +$   
 $Binnenlandse\_inmigratie\_BEV[31] + Geboorte\_BEV[31] - Emigratie\_BEV[31] -$   
 $Sterfte\_BEV[31] - Binnenlandse\_uitmigratie\_BEV[31]) * dt$   
 INIT BEVOLKINfG[31] = 103700  
 $BEVOLKINfG[32](t) = BEVOLKINfG[32](t - dt) + (Immigratie\_BEV[32] +$   
 $Binnenlandse\_inmigratie\_BEV[32] + Geboorte\_BEV[32] - Emigratie\_BEV[32] -$   
 $Sterfte\_BEV[32] - Binnenlandse\_uitmigratie\_BEV[32]) * dt$   
 INIT BEVOLKINfG[32] = 204500  
 $BEVOLKINfG[33](t) = BEVOLKINfG[33](t - dt) + (Immigratie\_BEV[33] +$   
 $Binnenlandse\_inmigratie\_BEV[33] + Geboorte\_BEV[33] - Emigratie\_BEV[33] -$   
 $Sterfte\_BEV[33] - Binnenlandse\_uitmigratie\_BEV[33]) * dt$

```

INIT BEVOLKINfG[33] = 455000
BEVOLKINfG[34](t) = BEVOLKINfG[34](t - dt) + (Immigratie_BEV[34] +
Binnenlandse_inmigratie_BEV[34] + Geboorte_BEV[34] - Emigratie_BEV[34] -
Sterfte_BEV[34] - Binnenlandse_uitmigratie_BEV[34]) * dt
INIT BEVOLKINfG[34] = 341900
BEVOLKINfG[35](t) = BEVOLKINfG[35](t - dt) + (Immigratie_BEV[35] +
Binnenlandse_inmigratie_BEV[35] + Geboorte_BEV[35] - Emigratie_BEV[35] -
Sterfte_BEV[35] - Binnenlandse_uitmigratie_BEV[35]) * dt
INIT BEVOLKINfG[35] = 462000
BEVOLKINfG[36](t) = BEVOLKINfG[36](t - dt) + (Immigratie_BEV[36] +
Binnenlandse_inmigratie_BEV[36] + Geboorte_BEV[36] - Emigratie_BEV[36] -
Sterfte_BEV[36] - Binnenlandse_uitmigratie_BEV[36]) * dt
INIT BEVOLKINfG[36] = 559500
BEVOLKINfG[37](t) = BEVOLKINfG[37](t - dt) + (Immigratie_BEV[37] +
Binnenlandse_inmigratie_BEV[37] + Geboorte_BEV[37] - Emigratie_BEV[37] -
Sterfte_BEV[37] - Binnenlandse_uitmigratie_BEV[37]) * dt
INIT BEVOLKINfG[37] = 225000
BEVOLKINfG[38](t) = BEVOLKINfG[38](t - dt) + (Immigratie_BEV[38] +
Binnenlandse_inmigratie_BEV[38] + Geboorte_BEV[38] - Emigratie_BEV[38] -
Sterfte_BEV[38] - Binnenlandse_uitmigratie_BEV[38]) * dt
INIT BEVOLKINfG[38] = 185000
BEVOLKINfG[39](t) = BEVOLKINfG[39](t - dt) + (Immigratie_BEV[39] +
Binnenlandse_inmigratie_BEV[39] + Geboorte_BEV[39] - Emigratie_BEV[39] -
Sterfte_BEV[39] - Binnenlandse_uitmigratie_BEV[39]) * dt
INIT BEVOLKINfG[39] = 604350
BEVOLKINfG[40](t) = BEVOLKINfG[40](t - dt) + (Immigratie_BEV[40] +
Binnenlandse_inmigratie_BEV[40] + Geboorte_BEV[40] - Emigratie_BEV[40] -
Sterfte_BEV[40] - Binnenlandse_uitmigratie_BEV[40]) * dt
INIT BEVOLKINfG[40] = 66500
Immigratie_BEV[van] = BEVOLKINfG[van]*pct_IMM_BEV[van]
Binnenlandse_inmigratie_BEV[van] = (RESINM[van]+ARBINM[van])
Geboorte_BEV[van] = BEVOLKINfG[van]*pct_GEB_BEV[van]
Emigratie_BEV[van] = BEVOLKINfG[van]*pct_EM_BEV[van]
Sterfte_BEV[van] = BEVOLKINfG[van]*pct_STERF_BEV[van]
Binnenlandse_uitmigratie_BEV[van] = (RESUITM[van]+ARBUITM[van])
ARBINM[van] = ARBINM_BEV[van]*(huishoudensgrootte[van]-
1)*huishoudensontwikkeling[van]
ARBUITM[van] = ARBUITM_BEV[van]*(huishoudensgrootte[van]-
1)*huishoudensontwikkeling[van]
EMF[van] = ABM_EM_BEV[van]*AWONM_EM_BEV[van]
Huishoudens[van] =
BEVOLKINfG[van]/(huishoudensgrootte[van]*huishoudensontwikkeling[van])
IMMF[van] = AWONM_IMM_BEV[van]*ABM_IMM_BEV[van]
pct_EM_BEV[van] = TIME
pct_GEB_BEV[van] = TIME
pct_IMM_BEV[van] = TIME
pct_STERF_BEV[van] = TIME
pot_RESINM[van] =
WONINGVOORRAAD[van]*AWONM_RESINM[van]*pct_RESINM[van]
pot_RESUITM[van] = BEVOLKINfG[van]*pct_RESUITM[van]*AWONM_RESUITM[van]
RESINM[van] = RESINM_BEV[van]
RESUITM[van] = RESUITM_BEV[van]
AWONM_RESINM[van] = GRAPH(HHWR[van])
(0.00, 1.71), (10.0, 1.70), (20.0, 1.66), (30.0, 1.58), (40.0, 1.38), (50.0, 1.00), (60.0, 0.6), (70.0,
0.43), (80.0, 0.36), (90.0, 0.3), (100, 0.28)

```



AWONM\_RESUITM[van] = GRAPH(HHWR[van])  
(0.00, 0.32), (0.2, 0.33), (0.4, 0.36), (0.6, 0.44), (0.8, 0.59), (1.00, 1.00), (1.20, 1.38), (1.40, 1.55),  
(1.60, 1.64), (1.80, 1.67), (2.00, 1.67)

huishoudensontwikkeling[van] = GRAPH(TIME)  
(1970, 1.03), (1971, 0.99), (1972, 0.96), (1973, 0.95), (1974, 0.93), (1975, 0.91), (1976, 0.89),  
(1977, 0.88), (1978, 0.86), (1979, 0.85), (1980, 0.84), (1981, 0.83), (1982, 0.81), (1983, 0.8),  
(1984, 0.79), (1985, 0.77), (1986, 0.75), (1987, 0.73), (1988, 0.72), (1989, 0.7), (1990, 0.69),  
(1991, 0.68), (1992, 0.67), (1993, 0.66), (1994, 0.66), (1995, 0.65), (1996, 0.65), (1997, 0.65),  
(1998, 0.65), (1999, 0.65), (2000, 0.65), (2001, 0.64), (2002, 0.64), (2003, 0.64), (2004, 0.64),  
(2005, 0.64), (2006, 0.63), (2007, 0.63), (2008, 0.62), (2009, 0.62), (2010, 0.62), (2011, 0.62),  
(2012, 0.61), (2013, 0.61), (2014, 0.61), (2015, 0.61), (2016, 0.6), (2017, 0.6), (2018, 0.6), (2019,  
0.6), (2020, 0.59), (2021, 0.59), (2022, 0.59), (2023, 0.59), (2024, 0.58), (2025, 0.58), (2026,  
0.58), (2027, 0.58), (2028, 0.57), (2029, 0.57), (2030, 0.57)

pct\_EM\_BEV[van] = TIME  
pct\_GEB\_BEV[van] = TIME  
pct\_IMM\_BEV[van] = TIME  
pct\_STERF\_BEV[van] = TIME

ABM\_INARB\_j[1] = ABM\_INARB[1]  
ABM\_INARB\_j[2] = ABM\_INARB[2]  
ABM\_INARB\_j[3] = ABM\_INARB[3]  
ABM\_INARB\_j[4] = ABM\_INARB[4]  
ABM\_INARB\_j[5] = ABM\_INARB[5]  
ABM\_INARB\_j[6] = ABM\_INARB[6]  
ABM\_INARB\_j[7] = ABM\_INARB[7]  
ABM\_INARB\_j[8] = ABM\_INARB[8]  
ABM\_INARB\_j[9] = ABM\_INARB[9]  
ABM\_INARB\_j[10] = ABM\_INARB[10]  
ABM\_INARB\_j[11] = ABM\_INARB[11]  
ABM\_INARB\_j[12] = ABM\_INARB[12]  
ABM\_INARB\_j[13] = ABM\_INARB[13]  
ABM\_INARB\_j[14] = ABM\_INARB[14]  
ABM\_INARB\_j[15] = ABM\_INARB[15]  
ABM\_INARB\_j[16] = ABM\_INARB[16]  
ABM\_INARB\_j[17] = ABM\_INARB[17]  
ABM\_INARB\_j[18] = ABM\_INARB[18]  
ABM\_INARB\_j[19] = ABM\_INARB[19]  
ABM\_INARB\_j[20] = ABM\_INARB[20]  
ABM\_INARB\_j[21] = ABM\_INARB[21]  
ABM\_INARB\_j[22] = ABM\_INARB[22]  
ABM\_INARB\_j[23] = ABM\_INARB[23]  
ABM\_INARB\_j[24] = ABM\_INARB[24]  
ABM\_INARB\_j[25] = ABM\_INARB[25]  
ABM\_INARB\_j[26] = ABM\_INARB[26]  
ABM\_INARB\_j[27] = ABM\_INARB[27]  
ABM\_INARB\_j[28] = ABM\_INARB[28]  
ABM\_INARB\_j[29] = ABM\_INARB[29]  
ABM\_INARB\_j[30] = ABM\_INARB[30]  
ABM\_INARB\_j[31] = ABM\_INARB[31]  
ABM\_INARB\_j[32] = ABM\_INARB[32]  
ABM\_INARB\_j[33] = ABM\_INARB[33]  
ABM\_INARB\_j[34] = ABM\_INARB[34]  
ABM\_INARB\_j[35] = ABM\_INARB[35]  
ABM\_INARB\_j[36] = ABM\_INARB[36]  
ABM\_INARB\_j[37] = ABM\_INARB[37]  
ABM\_INARB\_j[38] = ABM\_INARB[38]

```

ABM_INARB_j[39] = ABM_INARB[39]
ABM_INARB_j[40] = ABM_INARB[40]
ARBINM_BEV[1] = ARBINM_j[1]-ARBM_intra[1]
ARBINM_BEV[2] = ARBINM_j[2]-ARBM_intra[2]
ARBINM_BEV[3] = ARBINM_j[3]-ARBM_intra[3]
ARBINM_BEV[4] = ARBINM_j[4]-ARBM_intra[4]
ARBINM_BEV[5] = ARBINM_j[5]-ARBM_intra[5]
ARBINM_BEV[6] = ARBINM_j[6]-ARBM_intra[6]
ARBINM_BEV[7] = ARBINM_j[7]-ARBM_intra[7]
ARBINM_BEV[8] = ARBINM_j[8]-ARBM_intra[8]
ARBINM_BEV[9] = ARBINM_j[9]-ARBM_intra[9]
ARBINM_BEV[10] = ARBINM_j[10]-ARBM_intra[10]
ARBINM_BEV[11] = ARBINM_j[11]-ARBM_intra[11]
ARBINM_BEV[12] = ARBINM_j[12]-ARBM_intra[12]
ARBINM_BEV[13] = ARBINM_j[13]-ARBM_intra[13]
ARBINM_BEV[14] = ARBINM_j[14]-ARBM_intra[14]
ARBINM_BEV[15] = ARBINM_j[15]-ARBM_intra[15]
ARBINM_BEV[16] = ARBINM_j[16]-ARBM_intra[16]
ARBINM_BEV[17] = ARBINM_j[17]-ARBM_intra[17]
ARBINM_BEV[18] = ARBINM_j[18]-ARBM_intra[18]
ARBINM_BEV[19] = ARBINM_j[19]-ARBM_intra[19]
ARBINM_BEV[20] = ARBINM_j[20]-ARBM_intra[20]
ARBINM_BEV[21] = ARBINM_j[21]-ARBM_intra[21]
ARBINM_BEV[22] = ARBINM_j[22]-ARBM_intra[22]
ARBINM_BEV[23] = ARBINM_j[23]-ARBM_intra[23]
ARBINM_BEV[24] = ARBINM_j[24]-ARBM_intra[24]
ARBINM_BEV[25] = ARBINM_j[25]-ARBM_intra[25]
ARBINM_BEV[26] = ARBINM_j[26]-ARBM_intra[26]
ARBINM_BEV[27] = ARBINM_j[27]-ARBM_intra[27]
ARBINM_BEV[28] = ARBINM_j[28]-ARBM_intra[28]
ARBINM_BEV[29] = ARBINM_j[29]-ARBM_intra[29]
ARBINM_BEV[30] = ARBINM_j[30]-ARBM_intra[30]
ARBINM_BEV[31] = ARBINM_j[31]-ARBM_intra[31]
ARBINM_BEV[32] = ARBINM_j[32]-ARBM_intra[32]
ARBINM_BEV[33] = ARBINM_j[33]-ARBM_intra[33]
ARBINM_BEV[34] = ARBINM_j[34]-ARBM_intra[34]
ARBINM_BEV[35] = ARBINM_j[35]-ARBM_intra[35]
ARBINM_BEV[36] = ARBINM_j[36]-ARBM_intra[36]
ARBINM_BEV[37] = ARBINM_j[37]-ARBM_intra[37]
ARBINM_BEV[38] = ARBINM_j[38]-ARBM_intra[38]
ARBINM_BEV[39] = ARBINM_j[39]-ARBM_intra[39]
ARBINM_BEV[40] = ARBINM_j[40]-ARBM_intra[40]
ARBINM_j[naar] = ARRAYSUM(ARBM_ij[*],naar)
ARBMP_ij[van,naar] = pot_UITARB[van]*(BI_ARBMP_ij[van,naar]/P_ARBMP[van])
ARBM_ij[van,naar] = ARBMP_ij[van,naar]-Pendel_ij[van,naar]
ARBM_intra[1] = ARBM_ij[1,1]
ARBM_intra[2] = ARBM_ij[2,2]
ARBM_intra[3] = ARBM_ij[3,3]
ARBM_intra[4] = ARBM_ij[4,4]
ARBM_intra[5] = ARBM_ij[5,5]
ARBM_intra[6] = ARBM_ij[6,6]
ARBM_intra[7] = ARBM_ij[7,7]
ARBM_intra[8] = ARBM_ij[8,8]
ARBM_intra[9] = ARBM_ij[9,9]
ARBM_intra[10] = ARBM_ij[10,10]

```

```

ARBM_intra[11] = ARBM_ij[11,11]
ARBM_intra[12] = ARBM_ij[12,12]
ARBM_intra[13] = ARBM_ij[13,13]
ARBM_intra[14] = ARBM_ij[14,14]
ARBM_intra[15] = ARBM_ij[15,15]
ARBM_intra[16] = ARBM_ij[16,16]
ARBM_intra[17] = ARBM_ij[17,17]
ARBM_intra[18] = ARBM_ij[18,18]
ARBM_intra[19] = ARBM_ij[19,19]
ARBM_intra[20] = ARBM_ij[20,20]
ARBM_intra[21] = ARBM_ij[21,21]
ARBM_intra[22] = ARBM_ij[22,22]
ARBM_intra[23] = ARBM_ij[23,23]
ARBM_intra[24] = ARBM_ij[24,24]
ARBM_intra[25] = ARBM_ij[25,25]
ARBM_intra[26] = ARBM_ij[26,26]
ARBM_intra[27] = ARBM_ij[27,27]
ARBM_intra[28] = ARBM_ij[28,28]
ARBM_intra[29] = ARBM_ij[29,29]
ARBM_intra[30] = ARBM_ij[30,30]
ARBM_intra[31] = ARBM_ij[31,31]
ARBM_intra[32] = ARBM_ij[32,32]
ARBM_intra[33] = ARBM_ij[33,33]
ARBM_intra[34] = ARBM_ij[34,34]
ARBM_intra[35] = ARBM_ij[35,35]
ARBM_intra[36] = ARBM_ij[36,36]
ARBM_intra[37] = ARBM_ij[37,37]
ARBM_intra[38] = ARBM_ij[38,38]
ARBM_intra[39] = ARBM_ij[39,39]
ARBM_intra[40] = ARBM_ij[40,40]
ARBUITM_BEV[van] = ARRAYSUM(ARBM_ij[van,*])-ARBM_intra[van]
BI_ARBMP_ij[van,naar] =
pot_INARB_j[naar]*correctiefactor_INARB_ij[van,naar]/reistijd[van,naar]^parameter_b
INP[1] = INP__j[1]-PEN_intra[1]
INP[2] = INP__j[2]-PEN_intra[2]
INP[3] = INP__j[3]-PEN_intra[3]
INP[4] = INP__j[4]-PEN_intra[4]
INP[5] = INP__j[5]-PEN_intra[5]
INP[6] = INP__j[6]-PEN_intra[6]
INP[7] = INP__j[7]-PEN_intra[7]
INP[8] = INP__j[8]-PEN_intra[8]
INP[9] = INP__j[9]-PEN_intra[9]
INP[10] = INP__j[10]-PEN_intra[10]
INP[11] = INP__j[11]-PEN_intra[11]
INP[12] = INP__j[12]-PEN_intra[12]
INP[13] = INP__j[13]-PEN_intra[13]
INP[14] = INP__j[14]-PEN_intra[14]
INP[15] = INP__j[15]-PEN_intra[15]
INP[16] = INP__j[16]-PEN_intra[16]
INP[17] = INP__j[17]-PEN_intra[17]
INP[18] = INP__j[18]-PEN_intra[18]
INP[19] = INP__j[19]-PEN_intra[19]
INP[20] = INP__j[20]-PEN_intra[20]
INP[21] = INP__j[21]-PEN_intra[21]
INP[22] = INP__j[22]-PEN_intra[22]

```

```
INP[23] = INP__j[23]-PEN_intra[23]
INP[24] = INP__j[24]-PEN_intra[24]
INP[25] = INP__j[25]-PEN_intra[25]
INP[26] = INP__j[26]-PEN_intra[26]
INP[27] = INP__j[27]-PEN_intra[27]
INP[28] = INP__j[28]-PEN_intra[28]
INP[29] = INP__j[29]-PEN_intra[29]
INP[30] = INP__j[30]-PEN_intra[30]
INP[31] = INP__j[31]-PEN_intra[31]
INP[32] = INP__j[32]-PEN_intra[32]
INP[33] = INP__j[33]-PEN_intra[33]
INP[34] = INP__j[34]-PEN_intra[34]
INP[35] = INP__j[35]-PEN_intra[35]
INP[36] = INP__j[36]-PEN_intra[36]
INP[37] = INP__j[37]-PEN_intra[37]
INP[38] = INP__j[38]-PEN_intra[38]
INP[39] = INP__j[39]-PEN_intra[39]
INP[40] = INP__j[40]-PEN_intra[40]
INP__j[naar] = ARRAYSUM(Pendel_ij[*],naar)
parameter_b = 2.7
Pendel_ij[van,naar] = ARBMP_ij[van,naar]*pct_Pendel[van,naar]
PEN_intra[1] = Pendel_ij[1,1]
PEN_intra[2] = Pendel_ij[2,2]
PEN_intra[3] = Pendel_ij[3,3]
PEN_intra[4] = Pendel_ij[4,4]
PEN_intra[5] = Pendel_ij[5,5]
PEN_intra[6] = Pendel_ij[6,6]
PEN_intra[7] = Pendel_ij[7,7]
PEN_intra[8] = Pendel_ij[8,8]
PEN_intra[9] = Pendel_ij[9,9]
PEN_intra[10] = Pendel_ij[10,10]
PEN_intra[11] = Pendel_ij[11,11]
PEN_intra[12] = Pendel_ij[12,12]
PEN_intra[13] = Pendel_ij[13,13]
PEN_intra[14] = Pendel_ij[14,14]
PEN_intra[15] = Pendel_ij[15,15]
PEN_intra[16] = Pendel_ij[16,16]
PEN_intra[17] = Pendel_ij[17,17]
PEN_intra[18] = Pendel_ij[18,18]
PEN_intra[19] = Pendel_ij[19,19]
PEN_intra[20] = Pendel_ij[20,20]
PEN_intra[21] = Pendel_ij[21,21]
PEN_intra[22] = Pendel_ij[22,22]
PEN_intra[23] = Pendel_ij[23,23]
PEN_intra[24] = Pendel_ij[24,24]
PEN_intra[25] = Pendel_ij[25,25]
PEN_intra[26] = Pendel_ij[26,26]
PEN_intra[27] = Pendel_ij[27,27]
PEN_intra[28] = Pendel_ij[28,28]
PEN_intra[29] = Pendel_ij[29,29]
PEN_intra[30] = Pendel_ij[30,30]
PEN_intra[31] = Pendel_ij[31,31]
PEN_intra[32] = Pendel_ij[32,32]
PEN_intra[33] = Pendel_ij[33,33]
PEN_intra[34] = Pendel_ij[34,34]
```

```

PEN_intra[35] = Pendel_ij[35,35]
PEN_intra[36] = Pendel_ij[36,36]
PEN_intra[37] = Pendel_ij[37,37]
PEN_intra[38] = Pendel_ij[38,38]
PEN_intra[39] = Pendel_ij[39,39]
PEN_intra[40] = Pendel_ij[40,40]
pot_INARB_j[1] = pot_INARB[1]
pot_INARB_j[2] = pot_INARB[2]
pot_INARB_j[3] = pot_INARB[3]
pot_INARB_j[4] = pot_INARB[4]
pot_INARB_j[5] = pot_INARB[5]
pot_INARB_j[6] = pot_INARB[6]
pot_INARB_j[7] = pot_INARB[7]
pot_INARB_j[8] = pot_INARB[8]
pot_INARB_j[9] = pot_INARB[9]
pot_INARB_j[10] = pot_INARB[10]
pot_INARB_j[11] = pot_INARB[11]
pot_INARB_j[12] = pot_INARB[12]
pot_INARB_j[13] = pot_INARB[13]
pot_INARB_j[14] = pot_INARB[14]
pot_INARB_j[15] = pot_INARB[15]
pot_INARB_j[16] = pot_INARB[16]
pot_INARB_j[17] = pot_INARB[17]
pot_INARB_j[18] = pot_INARB[18]
pot_INARB_j[19] = pot_INARB[19]
pot_INARB_j[20] = pot_INARB[20]
pot_INARB_j[21] = pot_INARB[21]
pot_INARB_j[22] = pot_INARB[22]
pot_INARB_j[23] = pot_INARB[23]
pot_INARB_j[24] = pot_INARB[24]
pot_INARB_j[25] = pot_INARB[25]
pot_INARB_j[26] = pot_INARB[26]
pot_INARB_j[27] = pot_INARB[27]
pot_INARB_j[28] = pot_INARB[28]
pot_INARB_j[29] = pot_INARB[29]
pot_INARB_j[30] = pot_INARB[30]
pot_INARB_j[31] = pot_INARB[31]
pot_INARB_j[32] = pot_INARB[32]
pot_INARB_j[33] = pot_INARB[33]
pot_INARB_j[34] = pot_INARB[34]
pot_INARB_j[35] = pot_INARB[35]
pot_INARB_j[36] = pot_INARB[36]
pot_INARB_j[37] = pot_INARB[37]
pot_INARB_j[38] = pot_INARB[38]
pot_INARB_j[39] = pot_INARB[39]
pot_INARB_j[40] = pot_INARB[40]
P_ARBMP[van] = ARRAYSUM(BI_ARBMP_ij[van,*])
UITP[van] = ARRAYSUM(Pendel_ij[van,*])-PEN_intra[van]
correctiefactor_INARB_ij[van,naar] = GRAPH(ABM_INARB_j[naar]-ABM_INARB[van])
(-2.00, 0.11), (-1.60, 0.12), (-1.20, 0.17), (-0.8, 0.26), (-0.4, 0.48), (-1.11e-016, 1.00), (0.4, 1.58),
(0.8, 2.19), (1.20, 2.65), (1.60, 2.91), (2.00, 2.98)
BI_MIG_BV_ij[van,naar] =
pot_INM_BV_j[naar]*correctiefactor_MIG_BV_ij[van,naar]/reistijd[van,naar]^parameter_c
INMF_BV_j[1] = INMF_BV[1]
INMF_BV_j[2] = INMF_BV[2]

```

INMF\_BV\_j[3] = INMF\_BV[3]  
INMF\_BV\_j[4] = INMF\_BV[4]  
INMF\_BV\_j[5] = INMF\_BV[5]  
INMF\_BV\_j[6] = INMF\_BV[6]  
INMF\_BV\_j[7] = INMF\_BV[7]  
INMF\_BV\_j[8] = INMF\_BV[8]  
INMF\_BV\_j[9] = INMF\_BV[9]  
INMF\_BV\_j[10] = INMF\_BV[10]  
INMF\_BV\_j[11] = INMF\_BV[11]  
INMF\_BV\_j[12] = INMF\_BV[12]  
INMF\_BV\_j[13] = INMF\_BV[13]  
INMF\_BV\_j[14] = INMF\_BV[14]  
INMF\_BV\_j[15] = INMF\_BV[15]  
INMF\_BV\_j[16] = INMF\_BV[16]  
INMF\_BV\_j[17] = INMF\_BV[17]  
INMF\_BV\_j[18] = INMF\_BV[18]  
INMF\_BV\_j[19] = INMF\_BV[19]  
INMF\_BV\_j[20] = INMF\_BV[20]  
INMF\_BV\_j[21] = INMF\_BV[21]  
INMF\_BV\_j[22] = INMF\_BV[22]  
INMF\_BV\_j[23] = INMF\_BV[23]  
INMF\_BV\_j[24] = INMF\_BV[24]  
INMF\_BV\_j[25] = INMF\_BV[25]  
INMF\_BV\_j[26] = INMF\_BV[26]  
INMF\_BV\_j[27] = INMF\_BV[27]  
INMF\_BV\_j[28] = INMF\_BV[28]  
INMF\_BV\_j[29] = INMF\_BV[29]  
INMF\_BV\_j[30] = INMF\_BV[30]  
INMF\_BV\_j[31] = INMF\_BV[31]  
INMF\_BV\_j[32] = INMF\_BV[32]  
INMF\_BV\_j[33] = INMF\_BV[33]  
INMF\_BV\_j[34] = INMF\_BV[34]  
INMF\_BV\_j[35] = INMF\_BV[35]  
INMF\_BV\_j[36] = INMF\_BV[36]  
INMF\_BV\_j[37] = INMF\_BV[37]  
INMF\_BV\_j[38] = INMF\_BV[38]  
INMF\_BV\_j[39] = INMF\_BV[39]  
INMF\_BV\_j[40] = INMF\_BV[40]  
INM\_BV[1] = INM\_BV\_j[1]-MIG\_BV\_intra[1]  
INM\_BV[2] = INM\_BV\_j[2]-MIG\_BV\_intra[2]  
INM\_BV[3] = INM\_BV\_j[3]-MIG\_BV\_intra[3]  
INM\_BV[4] = INM\_BV\_j[4]-MIG\_BV\_intra[4]  
INM\_BV[5] = INM\_BV\_j[5]-MIG\_BV\_intra[5]  
INM\_BV[6] = INM\_BV\_j[6]-MIG\_BV\_intra[6]  
INM\_BV[7] = INM\_BV\_j[7]-MIG\_BV\_intra[7]  
INM\_BV[8] = INM\_BV\_j[8]-MIG\_BV\_intra[8]  
INM\_BV[9] = INM\_BV\_j[9]-MIG\_BV\_intra[9]  
INM\_BV[10] = INM\_BV\_j[10]-MIG\_BV\_intra[10]  
INM\_BV[11] = INM\_BV\_j[11]-MIG\_BV\_intra[11]  
INM\_BV[12] = INM\_BV\_j[12]-MIG\_BV\_intra[12]  
INM\_BV[13] = INM\_BV\_j[13]-MIG\_BV\_intra[13]  
INM\_BV[14] = INM\_BV\_j[14]-MIG\_BV\_intra[14]  
INM\_BV[15] = INM\_BV\_j[15]-MIG\_BV\_intra[15]  
INM\_BV[16] = INM\_BV\_j[16]-MIG\_BV\_intra[16]  
INM\_BV[17] = INM\_BV\_j[17]-MIG\_BV\_intra[17]

```

INM_BV[18] = INM_BV_j[18]-MIG_BV_intra[18]
INM_BV[19] = INM_BV_j[19]-MIG_BV_intra[19]
INM_BV[20] = INM_BV_j[20]-MIG_BV_intra[20]
INM_BV[21] = INM_BV_j[21]-MIG_BV_intra[21]
INM_BV[22] = INM_BV_j[22]-MIG_BV_intra[22]
INM_BV[23] = INM_BV_j[23]-MIG_BV_intra[23]
INM_BV[24] = INM_BV_j[24]-MIG_BV_intra[24]
INM_BV[25] = INM_BV_j[25]-MIG_BV_intra[25]
INM_BV[26] = INM_BV_j[26]-MIG_BV_intra[26]
INM_BV[27] = INM_BV_j[27]-MIG_BV_intra[27]
INM_BV[28] = INM_BV_j[28]-MIG_BV_intra[28]
INM_BV[29] = INM_BV_j[29]-MIG_BV_intra[29]
INM_BV[30] = INM_BV_j[30]-MIG_BV_intra[30]
INM_BV[31] = INM_BV_j[31]-MIG_BV_intra[31]
INM_BV[32] = INM_BV_j[32]-MIG_BV_intra[32]
INM_BV[33] = INM_BV_j[33]-MIG_BV_intra[33]
INM_BV[34] = INM_BV_j[34]-MIG_BV_intra[34]
INM_BV[35] = INM_BV_j[35]-MIG_BV_intra[35]
INM_BV[36] = INM_BV_j[36]-MIG_BV_intra[36]
INM_BV[37] = INM_BV_j[37]-MIG_BV_intra[37]
INM_BV[38] = INM_BV_j[38]-MIG_BV_intra[38]
INM_BV[39] = INM_BV_j[39]-MIG_BV_intra[39]
INM_BV[40] = INM_BV_j[40]-MIG_BV_intra[40]
INM_BV_j[naar] = ARRAYSUM(MIG_BV_ij[*],naar)
MIG_BV_ij[van,naar] = pot_UTM_BV[van]*(BI_MIG_BV_ij[van,naar]/P_MIG_BV[van])
MIG_BV_intra[1] = MIG_BV_ij[1,1]
MIG_BV_intra[2] = MIG_BV_ij[2,2]
MIG_BV_intra[3] = MIG_BV_ij[3,3]
MIG_BV_intra[4] = MIG_BV_ij[4,4]
MIG_BV_intra[5] = MIG_BV_ij[5,5]
MIG_BV_intra[6] = MIG_BV_ij[6,6]
MIG_BV_intra[7] = MIG_BV_ij[7,7]
MIG_BV_intra[8] = MIG_BV_ij[8,8]
MIG_BV_intra[9] = MIG_BV_ij[9,9]
MIG_BV_intra[10] = MIG_BV_ij[10,10]
MIG_BV_intra[11] = MIG_BV_ij[11,11]
MIG_BV_intra[12] = MIG_BV_ij[12,12]
MIG_BV_intra[13] = MIG_BV_ij[13,13]
MIG_BV_intra[14] = MIG_BV_ij[14,14]
MIG_BV_intra[15] = MIG_BV_ij[15,15]
MIG_BV_intra[16] = MIG_BV_ij[16,16]
MIG_BV_intra[17] = MIG_BV_ij[17,17]
MIG_BV_intra[18] = MIG_BV_ij[18,18]
MIG_BV_intra[19] = MIG_BV_ij[19,19]
MIG_BV_intra[20] = MIG_BV_ij[20,20]
MIG_BV_intra[21] = MIG_BV_ij[21,21]
MIG_BV_intra[22] = MIG_BV_ij[22,22]
MIG_BV_intra[23] = MIG_BV_ij[23,23]
MIG_BV_intra[24] = MIG_BV_ij[24,24]
MIG_BV_intra[25] = MIG_BV_ij[25,25]
MIG_BV_intra[26] = MIG_BV_ij[26,26]
MIG_BV_intra[27] = MIG_BV_ij[27,27]
MIG_BV_intra[28] = MIG_BV_ij[28,28]
MIG_BV_intra[29] = MIG_BV_ij[29,29]
MIG_BV_intra[30] = MIG_BV_ij[30,30]

```

```

MIG_BV_intra[31] = MIG_BV_ij[31,31]
MIG_BV_intra[32] = MIG_BV_ij[32,32]
MIG_BV_intra[33] = MIG_BV_ij[33,33]
MIG_BV_intra[34] = MIG_BV_ij[34,34]
MIG_BV_intra[35] = MIG_BV_ij[35,35]
MIG_BV_intra[36] = MIG_BV_ij[36,36]
MIG_BV_intra[37] = MIG_BV_ij[37,37]
MIG_BV_intra[38] = MIG_BV_ij[38,38]
MIG_BV_intra[39] = MIG_BV_ij[39,39]
MIG_BV_intra[40] = MIG_BV_ij[40,40]
parameter_c = 4
pot_INM_BV_j[1] = pot_INM_BV[1]
pot_INM_BV_j[2] = pot_INM_BV[2]
pot_INM_BV_j[3] = pot_INM_BV[3]
pot_INM_BV_j[4] = pot_INM_BV[4]
pot_INM_BV_j[5] = pot_INM_BV[5]
pot_INM_BV_j[6] = pot_INM_BV[6]
pot_INM_BV_j[7] = pot_INM_BV[7]
pot_INM_BV_j[8] = pot_INM_BV[8]
pot_INM_BV_j[9] = pot_INM_BV[9]
pot_INM_BV_j[10] = pot_INM_BV[10]
pot_INM_BV_j[11] = pot_INM_BV[11]
pot_INM_BV_j[12] = pot_INM_BV[12]
pot_INM_BV_j[13] = pot_INM_BV[13]
pot_INM_BV_j[14] = pot_INM_BV[14]
pot_INM_BV_j[15] = pot_INM_BV[15]
pot_INM_BV_j[16] = pot_INM_BV[16]
pot_INM_BV_j[17] = pot_INM_BV[17]
pot_INM_BV_j[18] = pot_INM_BV[18]
pot_INM_BV_j[19] = pot_INM_BV[19]
pot_INM_BV_j[20] = pot_INM_BV[20]
pot_INM_BV_j[21] = pot_INM_BV[21]
pot_INM_BV_j[22] = pot_INM_BV[22]
pot_INM_BV_j[23] = pot_INM_BV[23]
pot_INM_BV_j[24] = pot_INM_BV[24]
pot_INM_BV_j[25] = pot_INM_BV[25]
pot_INM_BV_j[26] = pot_INM_BV[26]
pot_INM_BV_j[27] = pot_INM_BV[27]
pot_INM_BV_j[28] = pot_INM_BV[28]
pot_INM_BV_j[29] = pot_INM_BV[29]
pot_INM_BV_j[30] = pot_INM_BV[30]
pot_INM_BV_j[31] = pot_INM_BV[31]
pot_INM_BV_j[32] = pot_INM_BV[32]
pot_INM_BV_j[33] = pot_INM_BV[33]
pot_INM_BV_j[34] = pot_INM_BV[34]
pot_INM_BV_j[35] = pot_INM_BV[35]
pot_INM_BV_j[36] = pot_INM_BV[36]
pot_INM_BV_j[37] = pot_INM_BV[37]
pot_INM_BV_j[38] = pot_INM_BV[38]
pot_INM_BV_j[39] = pot_INM_BV[39]
pot_INM_BV_j[40] = pot_INM_BV[40]
P_MIG_BV[van] = ARRAYSUM(BI_MIG_BV_ij[van,*])
UITM_BV[van] = ARRAYSUM(MIG_BV_ij[van,*])-MIG_BV_intra[van]
correctiefactor_MIG_BV_ij[van,naar] = GRAPH(INMF_BV_j[naar]-INMF_BV[van])

```



(-2.00, 0.11), (-1.60, 0.12), (-1.20, 0.16), (-0.8, 0.24), (-0.4, 0.495), (-1.11e-016, 1.00), (0.4, 1.58), (0.8, 2.19), (1.20, 2.65), (1.60, 2.91), (2.00, 2.98)

AWONM\_RESINM\_j[1] = AWONM\_RESINM[1]  
AWONM\_RESINM\_j[2] = AWONM\_RESINM[2]  
AWONM\_RESINM\_j[3] = AWONM\_RESINM[3]  
AWONM\_RESINM\_j[4] = AWONM\_RESINM[4]  
AWONM\_RESINM\_j[5] = AWONM\_RESINM[5]  
AWONM\_RESINM\_j[6] = AWONM\_RESINM[6]  
AWONM\_RESINM\_j[7] = AWONM\_RESINM[7]  
AWONM\_RESINM\_j[8] = AWONM\_RESINM[8]  
AWONM\_RESINM\_j[9] = AWONM\_RESINM[9]  
AWONM\_RESINM\_j[10] = AWONM\_RESINM[10]  
AWONM\_RESINM\_j[11] = AWONM\_RESINM[11]  
AWONM\_RESINM\_j[12] = AWONM\_RESINM[12]  
AWONM\_RESINM\_j[13] = AWONM\_RESINM[13]  
AWONM\_RESINM\_j[14] = AWONM\_RESINM[14]  
AWONM\_RESINM\_j[15] = AWONM\_RESINM[15]  
AWONM\_RESINM\_j[16] = AWONM\_RESINM[16]  
AWONM\_RESINM\_j[17] = AWONM\_RESINM[17]  
AWONM\_RESINM\_j[18] = AWONM\_RESINM[18]  
AWONM\_RESINM\_j[19] = AWONM\_RESINM[19]  
AWONM\_RESINM\_j[20] = AWONM\_RESINM[20]  
AWONM\_RESINM\_j[21] = AWONM\_RESINM[21]  
AWONM\_RESINM\_j[22] = AWONM\_RESINM[22]  
AWONM\_RESINM\_j[23] = AWONM\_RESINM[23]  
AWONM\_RESINM\_j[24] = AWONM\_RESINM[24]  
AWONM\_RESINM\_j[25] = AWONM\_RESINM[25]  
AWONM\_RESINM\_j[26] = AWONM\_RESINM[26]  
AWONM\_RESINM\_j[27] = AWONM\_RESINM[27]  
AWONM\_RESINM\_j[28] = AWONM\_RESINM[28]  
AWONM\_RESINM\_j[29] = AWONM\_RESINM[29]  
AWONM\_RESINM\_j[30] = AWONM\_RESINM[30]  
AWONM\_RESINM\_j[31] = AWONM\_RESINM[31]  
AWONM\_RESINM\_j[32] = AWONM\_RESINM[32]  
AWONM\_RESINM\_j[33] = AWONM\_RESINM[33]  
AWONM\_RESINM\_j[34] = AWONM\_RESINM[34]  
AWONM\_RESINM\_j[35] = AWONM\_RESINM[35]  
AWONM\_RESINM\_j[36] = AWONM\_RESINM[36]  
AWONM\_RESINM\_j[37] = AWONM\_RESINM[37]  
AWONM\_RESINM\_j[38] = AWONM\_RESINM[38]  
AWONM\_RESINM\_j[39] = AWONM\_RESINM[39]  
AWONM\_RESINM\_j[40] = AWONM\_RESINM[40]

BI\_RESM\_ij[van,naar] =  
pot\_RESINM\_j[naar]\*correctiefactor\_RESM\_ij[van,naar]/reistijd[van,naar]^parameter\_a  
parameter\_a = 2.9

pot\_RESINM\_j[1] = pot\_RESINM[1]  
pot\_RESINM\_j[2] = pot\_RESINM[2]  
pot\_RESINM\_j[3] = pot\_RESINM[3]  
pot\_RESINM\_j[4] = pot\_RESINM[4]  
pot\_RESINM\_j[5] = pot\_RESINM[5]  
pot\_RESINM\_j[6] = pot\_RESINM[6]  
pot\_RESINM\_j[7] = pot\_RESINM[7]  
pot\_RESINM\_j[8] = pot\_RESINM[8]  
pot\_RESINM\_j[9] = pot\_RESINM[9]  
pot\_RESINM\_j[10] = pot\_RESINM[10]

```
pot_RESINM_j[11] = pot_RESINM[11]
pot_RESINM_j[12] = pot_RESINM[12]
pot_RESINM_j[13] = pot_RESINM[13]
pot_RESINM_j[14] = pot_RESINM[14]
pot_RESINM_j[15] = pot_RESINM[15]
pot_RESINM_j[16] = pot_RESINM[16]
pot_RESINM_j[17] = pot_RESINM[17]
pot_RESINM_j[18] = pot_RESINM[18]
pot_RESINM_j[19] = pot_RESINM[19]
pot_RESINM_j[20] = pot_RESINM[20]
pot_RESINM_j[21] = pot_RESINM[21]
pot_RESINM_j[22] = pot_RESINM[22]
pot_RESINM_j[23] = pot_RESINM[23]
pot_RESINM_j[24] = pot_RESINM[24]
pot_RESINM_j[25] = pot_RESINM[25]
pot_RESINM_j[26] = pot_RESINM[26]
pot_RESINM_j[27] = pot_RESINM[27]
pot_RESINM_j[28] = pot_RESINM[28]
pot_RESINM_j[29] = pot_RESINM[29]
pot_RESINM_j[30] = pot_RESINM[30]
pot_RESINM_j[31] = pot_RESINM[31]
pot_RESINM_j[32] = pot_RESINM[32]
pot_RESINM_j[33] = pot_RESINM[33]
pot_RESINM_j[34] = pot_RESINM[34]
pot_RESINM_j[35] = pot_RESINM[35]
pot_RESINM_j[36] = pot_RESINM[36]
pot_RESINM_j[37] = pot_RESINM[37]
pot_RESINM_j[38] = pot_RESINM[38]
pot_RESINM_j[39] = pot_RESINM[39]
pot_RESINM_j[40] = pot_RESINM[40]
P_RESM[van] = ARRAYSUM(BI_RESM_ij[van,*])
RESINM_BEV[1] = RESINM_j[1]-RESM_intra[1]
RESINM_BEV[2] = RESINM_j[2]-RESM_intra[2]
RESINM_BEV[3] = RESINM_j[3]-RESM_intra[3]
RESINM_BEV[4] = RESINM_j[4]-RESM_intra[4]
RESINM_BEV[5] = RESINM_j[5]-RESM_intra[5]
RESINM_BEV[6] = RESINM_j[6]-RESM_intra[6]
RESINM_BEV[7] = RESINM_j[7]-RESM_intra[7]
RESINM_BEV[8] = RESINM_j[8]-RESM_intra[8]
RESINM_BEV[9] = RESINM_j[9]-RESM_intra[9]
RESINM_BEV[10] = RESINM_j[10]-RESM_intra[10]
RESINM_BEV[11] = RESINM_j[11]-RESM_intra[11]
RESINM_BEV[12] = RESINM_j[12]-RESM_intra[12]
RESINM_BEV[13] = RESINM_j[13]-RESM_intra[13]
RESINM_BEV[14] = RESINM_j[14]-RESM_intra[14]
RESINM_BEV[15] = RESINM_j[15]-RESM_intra[15]
RESINM_BEV[16] = RESINM_j[16]-RESM_intra[16]
RESINM_BEV[17] = RESINM_j[17]-RESM_intra[17]
RESINM_BEV[18] = RESINM_j[18]-RESM_intra[18]
RESINM_BEV[19] = RESINM_j[19]-RESM_intra[19]
RESINM_BEV[20] = RESINM_j[20]-RESM_intra[20]
RESINM_BEV[21] = RESINM_j[21]-RESM_intra[21]
RESINM_BEV[22] = RESINM_j[22]-RESM_intra[22]
RESINM_BEV[23] = RESINM_j[23]-RESM_intra[23]
RESINM_BEV[24] = RESINM_j[24]-RESM_intra[24]
```

```
RESINM_BEV[25] = RESINM_j[25]-RESM_intra[25]
RESINM_BEV[26] = RESINM_j[26]-RESM_intra[26]
RESINM_BEV[27] = RESINM_j[27]-RESM_intra[27]
RESINM_BEV[28] = RESINM_j[28]-RESM_intra[28]
RESINM_BEV[29] = RESINM_j[29]-RESM_intra[29]
RESINM_BEV[30] = RESINM_j[30]-RESM_intra[30]
RESINM_BEV[31] = RESINM_j[31]-RESM_intra[31]
RESINM_BEV[32] = RESINM_j[32]-RESM_intra[32]
RESINM_BEV[33] = RESINM_j[33]-RESM_intra[33]
RESINM_BEV[34] = RESINM_j[34]-RESM_intra[34]
RESINM_BEV[35] = RESINM_j[35]-RESM_intra[35]
RESINM_BEV[36] = RESINM_j[36]-RESM_intra[36]
RESINM_BEV[37] = RESINM_j[37]-RESM_intra[37]
RESINM_BEV[38] = RESINM_j[38]-RESM_intra[38]
RESINM_BEV[39] = RESINM_j[39]-RESM_intra[39]
RESINM_BEV[40] = RESINM_j[40]-RESM_intra[40]
RESINM_j[naar] = ARRAYSUM(RESM_ij[*],naar)
RESM_ij[van,naar] = pot_RESUITM[van]*(BI_RESM_ij[van,naar]/P_RESM[van])
RESM_intra[1] = RESM_ij[1,1]
RESM_intra[2] = RESM_ij[2,2]
RESM_intra[3] = RESM_ij[3,3]
RESM_intra[4] = RESM_ij[4,4]
RESM_intra[5] = RESM_ij[5,5]
RESM_intra[6] = RESM_ij[6,6]
RESM_intra[7] = RESM_ij[7,7]
RESM_intra[8] = RESM_ij[8,8]
RESM_intra[9] = RESM_ij[9,9]
RESM_intra[10] = RESM_ij[10,10]
RESM_intra[11] = RESM_ij[11,11]
RESM_intra[12] = RESM_ij[12,12]
RESM_intra[13] = RESM_ij[13,13]
RESM_intra[14] = RESM_ij[14,14]
RESM_intra[15] = RESM_ij[15,15]
RESM_intra[16] = RESM_ij[16,16]
RESM_intra[17] = RESM_ij[17,17]
RESM_intra[18] = RESM_ij[18,18]
RESM_intra[19] = RESM_ij[19,19]
RESM_intra[20] = RESM_ij[20,20]
RESM_intra[21] = RESM_ij[21,21]
RESM_intra[22] = RESM_ij[22,22]
RESM_intra[23] = RESM_ij[23,23]
RESM_intra[24] = RESM_ij[24,24]
RESM_intra[25] = RESM_ij[25,25]
RESM_intra[26] = RESM_ij[26,26]
RESM_intra[27] = RESM_ij[27,27]
RESM_intra[28] = RESM_ij[28,28]
RESM_intra[29] = RESM_ij[29,29]
RESM_intra[30] = RESM_ij[30,30]
RESM_intra[31] = RESM_ij[31,31]
RESM_intra[32] = RESM_ij[32,32]
RESM_intra[33] = RESM_ij[33,33]
RESM_intra[34] = RESM_ij[34,34]
RESM_intra[35] = RESM_ij[35,35]
RESM_intra[36] = RESM_ij[36,36]
RESM_intra[37] = RESM_ij[37,37]
```

```

RESM_intra[38] = RESM_ij[38,38]
RESM_intra[39] = RESM_ij[39,39]
RESM_intra[40] = RESM_ij[40,40]
RESUITM_BEV[van] = ARRAYSUM(RESM_ij[van,*])-RESM_intra[van]
correctiefactor_RESM_ij[van,naar] = GRAPH(AWONM_RESINM_j[naar]-
AWONM_RESINM[van])
(-2.00, 0.11), (-1.60, 0.12), (-1.20, 0.16), (-0.8, 0.24), (-0.4, 0.495), (-1.11e-016, 1.00), (0.4, 1.58),
(0.8, 2.19), (1.20, 2.65), (1.60, 2.91), (2.00, 2.98)
BL[van] =
(landoppervlak_bedrijfsgebied_in_ha[van]+landoppervlak_woongebied_in_ha[van])/(totaal_opp_
in_ha[van])
gem_opp_LAND_per_BV_in_m2[1] = 1480
gem_opp_LAND_per_BV_in_m2[2] = 2400
gem_opp_LAND_per_BV_in_m2[3] = 880
gem_opp_LAND_per_BV_in_m2[4] = 760
gem_opp_LAND_per_BV_in_m2[5] = 710
gem_opp_LAND_per_BV_in_m2[6] = 760
gem_opp_LAND_per_BV_in_m2[7] = 800
gem_opp_LAND_per_BV_in_m2[8] = 1150
gem_opp_LAND_per_BV_in_m2[9] = 830
gem_opp_LAND_per_BV_in_m2[10] = 910
gem_opp_LAND_per_BV_in_m2[11] = 820
gem_opp_LAND_per_BV_in_m2[12] = 960
gem_opp_LAND_per_BV_in_m2[13] = 720
gem_opp_LAND_per_BV_in_m2[14] = 800
gem_opp_LAND_per_BV_in_m2[15] = 870
gem_opp_LAND_per_BV_in_m2[16] = 1030
gem_opp_LAND_per_BV_in_m2[17] = 560
gem_opp_LAND_per_BV_in_m2[18] = 600
gem_opp_LAND_per_BV_in_m2[19] = 550.0
gem_opp_LAND_per_BV_in_m2[20] = 1770
gem_opp_LAND_per_BV_in_m2[21] = 350.0
gem_opp_LAND_per_BV_in_m2[22] = 1060.0
gem_opp_LAND_per_BV_in_m2[23] = 580.0
gem_opp_LAND_per_BV_in_m2[24] = 340.0
gem_opp_LAND_per_BV_in_m2[25] = 500.0
gem_opp_LAND_per_BV_in_m2[26] = 200.0
gem_opp_LAND_per_BV_in_m2[27] = 540.0
gem_opp_LAND_per_BV_in_m2[28] = 560.0
gem_opp_LAND_per_BV_in_m2[29] = 1290.0
gem_opp_LAND_per_BV_in_m2[30] = 1050.0
gem_opp_LAND_per_BV_in_m2[31] = 1680.0
gem_opp_LAND_per_BV_in_m2[32] = 1020.0
gem_opp_LAND_per_BV_in_m2[33] = 1230.0
gem_opp_LAND_per_BV_in_m2[34] = 860.0
gem_opp_LAND_per_BV_in_m2[35] = 880.0
gem_opp_LAND_per_BV_in_m2[36] = 910.0
gem_opp_LAND_per_BV_in_m2[37] = 970.0
gem_opp_LAND_per_BV_in_m2[38] = 950.0
gem_opp_LAND_per_BV_in_m2[39] = 1050.0
gem_opp_LAND_per_BV_in_m2[40] = 870.0
gem_opp_LAND_per_WON_in_m2[1] = 634.9
gem_opp_LAND_per_WON_in_m2[2] = 546.6
gem_opp_LAND_per_WON_in_m2[3] = 395.6
gem_opp_LAND_per_WON_in_m2[4] = 477.9

```

gem\_opp\_LAND\_per\_WON\_in\_m2[5] = 453.5  
gem\_opp\_LAND\_per\_WON\_in\_m2[6] = 484.9  
gem\_opp\_LAND\_per\_WON\_in\_m2[7] = 608.0  
gem\_opp\_LAND\_per\_WON\_in\_m2[8] = 619.1  
gem\_opp\_LAND\_per\_WON\_in\_m2[9] = 563.3  
gem\_opp\_LAND\_per\_WON\_in\_m2[10] = 423.7  
gem\_opp\_LAND\_per\_WON\_in\_m2[11] = 344.8  
gem\_opp\_LAND\_per\_WON\_in\_m2[12] = 405.0  
gem\_opp\_LAND\_per\_WON\_in\_m2[13] = 455.9  
gem\_opp\_LAND\_per\_WON\_in\_m2[14] = 455.3  
gem\_opp\_LAND\_per\_WON\_in\_m2[15] = 385.1  
gem\_opp\_LAND\_per\_WON\_in\_m2[16] = 484.6  
gem\_opp\_LAND\_per\_WON\_in\_m2[17] = 352.7  
gem\_opp\_LAND\_per\_WON\_in\_m2[18] = 389.7  
gem\_opp\_LAND\_per\_WON\_in\_m2[19] = 409.7  
gem\_opp\_LAND\_per\_WON\_in\_m2[20] = 289.1  
gem\_opp\_LAND\_per\_WON\_in\_m2[21] = 298.3  
gem\_opp\_LAND\_per\_WON\_in\_m2[22] = 277.3  
gem\_opp\_LAND\_per\_WON\_in\_m2[23] = 203.4  
gem\_opp\_LAND\_per\_WON\_in\_m2[24] = 411.9  
gem\_opp\_LAND\_per\_WON\_in\_m2[25] = 292.7  
gem\_opp\_LAND\_per\_WON\_in\_m2[26] = 189.6  
gem\_opp\_LAND\_per\_WON\_in\_m2[27] = 224.3  
gem\_opp\_LAND\_per\_WON\_in\_m2[28] = 309.0  
gem\_opp\_LAND\_per\_WON\_in\_m2[29] = 231.0  
gem\_opp\_LAND\_per\_WON\_in\_m2[30] = 300.1  
gem\_opp\_LAND\_per\_WON\_in\_m2[31] = 474.8  
gem\_opp\_LAND\_per\_WON\_in\_m2[32] = 427.3  
gem\_opp\_LAND\_per\_WON\_in\_m2[33] = 411.4  
gem\_opp\_LAND\_per\_WON\_in\_m2[34] = 476.5  
gem\_opp\_LAND\_per\_WON\_in\_m2[35] = 442.5  
gem\_opp\_LAND\_per\_WON\_in\_m2[36] = 475.9  
gem\_opp\_LAND\_per\_WON\_in\_m2[37] = 462.0  
gem\_opp\_LAND\_per\_WON\_in\_m2[38] = 544.2  
gem\_opp\_LAND\_per\_WON\_in\_m2[39] = 424.2  
gem\_opp\_LAND\_per\_WON\_in\_m2[40] = 398.4  
landoppervlak\_bedrijfsgebied\_in\_ha[van] =  
BEDRIJFSRUIMTEVOORRAAD[van]\*gem\_opp\_LAND\_per\_BV\_in\_m2[van]\*0.0001  
landoppervlak\_woongebied\_in\_ha[van] =  
WONINGVOORRAAD[van]\*gem\_opp\_LAND\_per\_WON\_in\_m2[van]\*0.0001  
totaal\_opp\_in\_ha[1] = 79770  
totaal\_opp\_in\_ha[2] = 24037  
totaal\_opp\_in\_ha[3] = 129725  
totaal\_opp\_in\_ha[4] = 147883  
totaal\_opp\_in\_ha[5] = 77901  
totaal\_opp\_in\_ha[6] = 109370  
totaal\_opp\_in\_ha[7] = 89491  
totaal\_opp\_in\_ha[8] = 79522  
totaal\_opp\_in\_ha[9] = 96344  
totaal\_opp\_in\_ha[10] = 198797  
totaal\_opp\_in\_ha[11] = 39466  
totaal\_opp\_in\_ha[12] = 142862  
totaal\_opp\_in\_ha[13] = 183683  
totaal\_opp\_in\_ha[14] = 151933  
totaal\_opp\_in\_ha[15] = 94494

totaal\_opp\_in\_ha[16] = 70638  
totaal\_opp\_in\_ha[17] = 133149  
totaal\_opp\_in\_ha[18] = 106061  
totaal\_opp\_in\_ha[19] = 31326  
totaal\_opp\_in\_ha[20] = 14172  
totaal\_opp\_in\_ha[21] = 13658  
totaal\_opp\_in\_ha[22] = 9550  
totaal\_opp\_in\_ha[23] = 73716  
totaal\_opp\_in\_ha[24] = 18096  
totaal\_opp\_in\_ha[25] = 23954  
totaal\_opp\_in\_ha[26] = 22066  
totaal\_opp\_in\_ha[27] = 18867  
totaal\_opp\_in\_ha[28] = 52190  
totaal\_opp\_in\_ha[29] = 120659  
totaal\_opp\_in\_ha[30] = 52910  
totaal\_opp\_in\_ha[31] = 73365  
totaal\_opp\_in\_ha[32] = 105171  
totaal\_opp\_in\_ha[33] = 128766  
totaal\_opp\_in\_ha[34] = 101847  
totaal\_opp\_in\_ha[35] = 128798  
totaal\_opp\_in\_ha[36] = 136195  
totaal\_opp\_in\_ha[37] = 84022  
totaal\_opp\_in\_ha[38] = 64802  
totaal\_opp\_in\_ha[39] = 68155  
totaal\_opp\_in\_ha[40] = 94905  
reistijd[van,naar] = reistijdenmatrix[van,naar]\*reistijdontwikkeling  
reistijdenmatrix[1,1] = 18  
reistijdenmatrix[1,2] = 32.4  
reistijdenmatrix[1,3] = 39.3  
reistijdenmatrix[1,4] = 70.6  
reistijdenmatrix[1,5] = 83.2  
reistijdenmatrix[1,6] = 70.7  
reistijdenmatrix[1,7] = 34  
reistijdenmatrix[1,8] = 45.3  
reistijdenmatrix[1,9] = 48.2  
reistijdenmatrix[1,10] = 69.5  
reistijdenmatrix[1,11] = 93.6  
reistijdenmatrix[1,12] = 91.1  
reistijdenmatrix[1,13] = 102.9  
reistijdenmatrix[1,14] = 128.4  
reistijdenmatrix[1,15] = 127.8  
reistijdenmatrix[1,16] = 145.9  
reistijdenmatrix[1,17] = 127.2  
reistijdenmatrix[1,18] = 121.9  
reistijdenmatrix[1,19] = 144.7  
reistijdenmatrix[1,20] = 159  
reistijdenmatrix[1,21] = 157.8  
reistijdenmatrix[1,22] = 148.4  
reistijdenmatrix[1,23] = 150.9  
reistijdenmatrix[1,24] = 125.6  
reistijdenmatrix[1,25] = 159.8  
reistijdenmatrix[1,26] = 165  
reistijdenmatrix[1,27] = 171.7  
reistijdenmatrix[1,28] = 154.8  
reistijdenmatrix[1,29] = 177.9

reistijdenmatrix[1,30] = 152.3  
reistijdenmatrix[1,31] = 257.4  
reistijdenmatrix[1,32] = 217.5  
reistijdenmatrix[1,33] = 179.7  
reistijdenmatrix[1,34] = 172.7  
reistijdenmatrix[1,35] = 141.6  
reistijdenmatrix[1,36] = 178  
reistijdenmatrix[1,37] = 161.8  
reistijdenmatrix[1,38] = 191.3  
reistijdenmatrix[1,39] = 220  
reistijdenmatrix[1,40] = 112.2  
reistijdenmatrix[2,1] = 32.4  
reistijdenmatrix[2,2] = 12  
reistijdenmatrix[2,3] = 20.6  
reistijdenmatrix[2,4] = 62.6  
reistijdenmatrix[2,5] = 75.1  
reistijdenmatrix[2,6] = 62.6  
reistijdenmatrix[2,7] = 41.3  
reistijdenmatrix[2,8] = 63.5  
reistijdenmatrix[2,9] = 57.2  
reistijdenmatrix[2,10] = 78.5  
reistijdenmatrix[2,11] = 102.6  
reistijdenmatrix[2,12] = 109.1  
reistijdenmatrix[2,13] = 111.8  
reistijdenmatrix[2,14] = 137.3  
reistijdenmatrix[2,15] = 136.7  
reistijdenmatrix[2,16] = 154.8  
reistijdenmatrix[2,17] = 136.1  
reistijdenmatrix[2,18] = 113.9  
reistijdenmatrix[2,19] = 136.7  
reistijdenmatrix[2,20] = 152.4  
reistijdenmatrix[2,21] = 151.2  
reistijdenmatrix[2,22] = 141.7  
reistijdenmatrix[2,23] = 144.2  
reistijdenmatrix[2,24] = 128.9  
reistijdenmatrix[2,25] = 153.1  
reistijdenmatrix[2,26] = 167.1  
reistijdenmatrix[2,27] = 174.6  
reistijdenmatrix[2,28] = 163.7  
reistijdenmatrix[2,29] = 186.9  
reistijdenmatrix[2,30] = 161.2  
reistijdenmatrix[2,31] = 266.4  
reistijdenmatrix[2,32] = 226.5  
reistijdenmatrix[2,33] = 188.7  
reistijdenmatrix[2,34] = 181.7  
reistijdenmatrix[2,35] = 150.5  
reistijdenmatrix[2,36] = 186.9  
reistijdenmatrix[2,37] = 170.8  
reistijdenmatrix[2,38] = 200.3  
reistijdenmatrix[2,39] = 228.9  
reistijdenmatrix[2,40] = 104.1  
reistijdenmatrix[3,1] = 39.3  
reistijdenmatrix[3,2] = 20.6  
reistijdenmatrix[3,3] = 19  
reistijdenmatrix[3,4] = 51.9

reistijdenmatrix[3,5] = 64.5  
reistijdenmatrix[3,6] = 52  
reistijdenmatrix[3,7] = 30.7  
reistijdenmatrix[3,8] = 56.2  
reistijdenmatrix[3,9] = 46.6  
reistijdenmatrix[3,10] = 67.9  
reistijdenmatrix[3,11] = 92  
reistijdenmatrix[3,12] = 98.5  
reistijdenmatrix[3,13] = 101.3  
reistijdenmatrix[3,14] = 126.8  
reistijdenmatrix[3,15] = 126.2  
reistijdenmatrix[3,16] = 144.3  
reistijdenmatrix[3,17] = 125.6  
reistijdenmatrix[3,18] = 103.2  
reistijdenmatrix[3,19] = 126  
reistijdenmatrix[3,20] = 141.8  
reistijdenmatrix[3,21] = 140.6  
reistijdenmatrix[3,22] = 131.1  
reistijdenmatrix[3,23] = 133.6  
reistijdenmatrix[3,24] = 118.3  
reistijdenmatrix[3,25] = 142.5  
reistijdenmatrix[3,26] = 156.5  
reistijdenmatrix[3,27] = 164  
reistijdenmatrix[3,28] = 153.2  
reistijdenmatrix[3,29] = 176.3  
reistijdenmatrix[3,30] = 150.6  
reistijdenmatrix[3,31] = 255.8  
reistijdenmatrix[3,32] = 215.9  
reistijdenmatrix[3,33] = 178.1  
reistijdenmatrix[3,34] = 171.1  
reistijdenmatrix[3,35] = 139.9  
reistijdenmatrix[3,36] = 176.4  
reistijdenmatrix[3,37] = 160.2  
reistijdenmatrix[3,38] = 189.7  
reistijdenmatrix[3,39] = 218.4  
reistijdenmatrix[3,40] = 93.5  
reistijdenmatrix[4,1] = 70.6  
reistijdenmatrix[4,2] = 62.6  
reistijdenmatrix[4,3] = 51.9  
reistijdenmatrix[4,4] = 13  
reistijdenmatrix[4,5] = 27.8  
reistijdenmatrix[4,6] = 30  
reistijdenmatrix[4,7] = 59.4  
reistijdenmatrix[4,8] = 71.3  
reistijdenmatrix[4,9] = 62  
reistijdenmatrix[4,10] = 58.1  
reistijdenmatrix[4,11] = 89.6  
reistijdenmatrix[4,12] = 106.1  
reistijdenmatrix[4,13] = 91.3  
reistijdenmatrix[4,14] = 126.7  
reistijdenmatrix[4,15] = 116.2  
reistijdenmatrix[4,16] = 122.9  
reistijdenmatrix[4,17] = 103.8  
reistijdenmatrix[4,18] = 57  
reistijdenmatrix[4,19] = 79.8



reistijdenmatrix[4,20] = 95.6  
reistijdenmatrix[4,21] = 103.4  
reistijdenmatrix[4,22] = 86.1  
reistijdenmatrix[4,23] = 97  
reistijdenmatrix[4,24] = 92.3  
reistijdenmatrix[4,25] = 111.9  
reistijdenmatrix[4,26] = 125.7  
reistijdenmatrix[4,27] = 133.1  
reistijdenmatrix[4,28] = 124.7  
reistijdenmatrix[4,29] = 149.5  
reistijdenmatrix[4,30] = 129.2  
reistijdenmatrix[4,31] = 234.3  
reistijdenmatrix[4,32] = 194.4  
reistijdenmatrix[4,33] = 156.7  
reistijdenmatrix[4,34] = 149.8  
reistijdenmatrix[4,35] = 129.9  
reistijdenmatrix[4,36] = 161.7  
reistijdenmatrix[4,37] = 150.2  
reistijdenmatrix[4,38] = 179.7  
reistijdenmatrix[4,39] = 204.2  
reistijdenmatrix[4,40] = 67.5  
reistijdenmatrix[5,1] = 83.2  
reistijdenmatrix[5,2] = 75.1  
reistijdenmatrix[5,3] = 64.5  
reistijdenmatrix[5,4] = 27.8  
reistijdenmatrix[5,5] = 10  
reistijdenmatrix[5,6] = 29.3  
reistijdenmatrix[5,7] = 71.3  
reistijdenmatrix[5,8] = 82.5  
reistijdenmatrix[5,9] = 61.7  
reistijdenmatrix[5,10] = 57.4  
reistijdenmatrix[5,11] = 88.9  
reistijdenmatrix[5,12] = 105.3  
reistijdenmatrix[5,13] = 90.3  
reistijdenmatrix[5,14] = 125.7  
reistijdenmatrix[5,15] = 115.2  
reistijdenmatrix[5,16] = 110  
reistijdenmatrix[5,17] = 91  
reistijdenmatrix[5,18] = 46.7  
reistijdenmatrix[5,19] = 69.5  
reistijdenmatrix[5,20] = 85.3  
reistijdenmatrix[5,21] = 93.2  
reistijdenmatrix[5,22] = 75.9  
reistijdenmatrix[5,23] = 86.7  
reistijdenmatrix[5,24] = 79.4  
reistijdenmatrix[5,25] = 101.7  
reistijdenmatrix[5,26] = 115.4  
reistijdenmatrix[5,27] = 122.9  
reistijdenmatrix[5,28] = 114.5  
reistijdenmatrix[5,29] = 139.3  
reistijdenmatrix[5,30] = 116.3  
reistijdenmatrix[5,31] = 221.4  
reistijdenmatrix[5,32] = 181.6  
reistijdenmatrix[5,33] = 143.9  
reistijdenmatrix[5,34] = 137

reistijdenmatrix[5,35] = 129  
reistijdenmatrix[5,36] = 148.9  
reistijdenmatrix[5,37] = 149.2  
reistijdenmatrix[5,38] = 167.6  
reistijdenmatrix[5,39] = 191.3  
reistijdenmatrix[5,40] = 54.7  
reistijdenmatrix[6,1] = 70.7  
reistijdenmatrix[6,2] = 62.6  
reistijdenmatrix[6,3] = 52  
reistijdenmatrix[6,4] = 30  
reistijdenmatrix[6,5] = 29.3  
reistijdenmatrix[6,6] = 14  
reistijdenmatrix[6,7] = 45  
reistijdenmatrix[6,8] = 56.2  
reistijdenmatrix[6,9] = 44.3  
reistijdenmatrix[6,10] = 41.3  
reistijdenmatrix[6,11] = 72.8  
reistijdenmatrix[6,12] = 89.3  
reistijdenmatrix[6,13] = 74.5  
reistijdenmatrix[6,14] = 109.9  
reistijdenmatrix[6,15] = 99.4  
reistijdenmatrix[6,16] = 113.5  
reistijdenmatrix[6,17] = 94.4  
reistijdenmatrix[6,18] = 68.1  
reistijdenmatrix[6,19] = 90.9  
reistijdenmatrix[6,20] = 106.4  
reistijdenmatrix[6,21] = 105.2  
reistijdenmatrix[6,22] = 95.7  
reistijdenmatrix[6,23] = 98.2  
reistijdenmatrix[6,24] = 82.9  
reistijdenmatrix[6,25] = 107.1  
reistijdenmatrix[6,26] = 121.1  
reistijdenmatrix[6,27] = 128.5  
reistijdenmatrix[6,28] = 120.1  
reistijdenmatrix[6,29] = 145  
reistijdenmatrix[6,30] = 119.8  
reistijdenmatrix[6,31] = 224.9  
reistijdenmatrix[6,32] = 185  
reistijdenmatrix[6,33] = 147.3  
reistijdenmatrix[6,34] = 140.4  
reistijdenmatrix[6,35] = 113.1  
reistijdenmatrix[6,36] = 149.6  
reistijdenmatrix[6,37] = 133.4  
reistijdenmatrix[6,38] = 162.9  
reistijdenmatrix[6,39] = 191.6  
reistijdenmatrix[6,40] = 58.1  
reistijdenmatrix[7,1] = 34  
reistijdenmatrix[7,2] = 41.3  
reistijdenmatrix[7,3] = 30.7  
reistijdenmatrix[7,4] = 59.4  
reistijdenmatrix[7,5] = 71.3  
reistijdenmatrix[7,6] = 45  
reistijdenmatrix[7,7] = 18  
reistijdenmatrix[7,8] = 31.9  
reistijdenmatrix[7,9] = 22.2

reistijdenmatrix[7,10] = 43.5  
reistijdenmatrix[7,11] = 67.6  
reistijdenmatrix[7,12] = 74.1  
reistijdenmatrix[7,13] = 76.9  
reistijdenmatrix[7,14] = 102.4  
reistijdenmatrix[7,15] = 101.8  
reistijdenmatrix[7,16] = 119.9  
reistijdenmatrix[7,17] = 101.2  
reistijdenmatrix[7,18] = 110.1  
reistijdenmatrix[7,19] = 132.9  
reistijdenmatrix[7,20] = 133  
reistijdenmatrix[7,21] = 131.8  
reistijdenmatrix[7,22] = 122.4  
reistijdenmatrix[7,23] = 124.9  
reistijdenmatrix[7,24] = 99.6  
reistijdenmatrix[7,25] = 133.8  
reistijdenmatrix[7,26] = 139  
reistijdenmatrix[7,27] = 145.7  
reistijdenmatrix[7,28] = 128.8  
reistijdenmatrix[7,29] = 151.9  
reistijdenmatrix[7,30] = 126.3  
reistijdenmatrix[7,31] = 231.4  
reistijdenmatrix[7,32] = 191.6  
reistijdenmatrix[7,33] = 153.8  
reistijdenmatrix[7,34] = 146.7  
reistijdenmatrix[7,35] = 115.6  
reistijdenmatrix[7,36] = 152  
reistijdenmatrix[7,37] = 135.8  
reistijdenmatrix[7,38] = 165.3  
reistijdenmatrix[7,39] = 194  
reistijdenmatrix[7,40] = 92.2  
reistijdenmatrix[8,1] = 45.3  
reistijdenmatrix[8,2] = 63.5  
reistijdenmatrix[8,3] = 56.2  
reistijdenmatrix[8,4] = 71.3  
reistijdenmatrix[8,5] = 82.5  
reistijdenmatrix[8,6] = 56.2  
reistijdenmatrix[8,7] = 31.9  
reistijdenmatrix[8,8] = 15  
reistijdenmatrix[8,9] = 32.8  
reistijdenmatrix[8,10] = 44.3  
reistijdenmatrix[8,11] = 64.1  
reistijdenmatrix[8,12] = 53.4  
reistijdenmatrix[8,13] = 79  
reistijdenmatrix[8,14] = 95.4  
reistijdenmatrix[8,15] = 103.9  
reistijdenmatrix[8,16] = 122  
reistijdenmatrix[8,17] = 103.6  
reistijdenmatrix[8,18] = 121.3  
reistijdenmatrix[8,19] = 144.1  
reistijdenmatrix[8,20] = 135.3  
reistijdenmatrix[8,21] = 134.1  
reistijdenmatrix[8,22] = 124.8  
reistijdenmatrix[8,23] = 127.2  
reistijdenmatrix[8,24] = 101.9

reistijdenmatrix[8,25] = 136.2  
reistijdenmatrix[8,26] = 141.1  
reistijdenmatrix[8,27] = 147.7  
reistijdenmatrix[8,28] = 131  
reistijdenmatrix[8,29] = 154  
reistijdenmatrix[8,30] = 128.3  
reistijdenmatrix[8,31] = 233.5  
reistijdenmatrix[8,32] = 193.6  
reistijdenmatrix[8,33] = 155.8  
reistijdenmatrix[8,34] = 148.8  
reistijdenmatrix[8,35] = 117.6  
reistijdenmatrix[8,36] = 154.1  
reistijdenmatrix[8,37] = 137.9  
reistijdenmatrix[8,38] = 167.4  
reistijdenmatrix[8,39] = 196.1  
reistijdenmatrix[8,40] = 94.6  
reistijdenmatrix[9,1] = 48.2  
reistijdenmatrix[9,2] = 57.2  
reistijdenmatrix[9,3] = 46.6  
reistijdenmatrix[9,4] = 62  
reistijdenmatrix[9,5] = 61.7  
reistijdenmatrix[9,6] = 44.3  
reistijdenmatrix[9,7] = 22.2  
reistijdenmatrix[9,8] = 32.8  
reistijdenmatrix[9,9] = 13  
reistijdenmatrix[9,10] = 28.4  
reistijdenmatrix[9,11] = 52.5  
reistijdenmatrix[9,12] = 59  
reistijdenmatrix[9,13] = 61.8  
reistijdenmatrix[9,14] = 87.3  
reistijdenmatrix[9,15] = 86.7  
reistijdenmatrix[9,16] = 104.8  
reistijdenmatrix[9,17] = 86.1  
reistijdenmatrix[9,18] = 100.4  
reistijdenmatrix[9,19] = 123.2  
reistijdenmatrix[9,20] = 117.9  
reistijdenmatrix[9,21] = 116.7  
reistijdenmatrix[9,22] = 107.3  
reistijdenmatrix[9,23] = 109.8  
reistijdenmatrix[9,24] = 84.5  
reistijdenmatrix[9,25] = 118.7  
reistijdenmatrix[9,26] = 123.9  
reistijdenmatrix[9,27] = 130.6  
reistijdenmatrix[9,28] = 113.7  
reistijdenmatrix[9,29] = 136.8  
reistijdenmatrix[9,30] = 111.2  
reistijdenmatrix[9,31] = 216.3  
reistijdenmatrix[9,32] = 176.5  
reistijdenmatrix[9,33] = 138.6  
reistijdenmatrix[9,34] = 131.6  
reistijdenmatrix[9,35] = 100.5  
reistijdenmatrix[9,36] = 136.9  
reistijdenmatrix[9,37] = 120.7  
reistijdenmatrix[9,38] = 150.2  
reistijdenmatrix[9,39] = 178.9

reistijdenmatrix[9,40] = 77.1  
reistijdenmatrix[10,1] = 69.5  
reistijdenmatrix[10,2] = 78.5  
reistijdenmatrix[10,3] = 67.9  
reistijdenmatrix[10,4] = 58.1  
reistijdenmatrix[10,5] = 57.4  
reistijdenmatrix[10,6] = 41.3  
reistijdenmatrix[10,7] = 43.5  
reistijdenmatrix[10,8] = 44.3  
reistijdenmatrix[10,9] = 28.4  
reistijdenmatrix[10,10] = 16  
reistijdenmatrix[10,11] = 40  
reistijdenmatrix[10,12] = 49.9  
reistijdenmatrix[10,13] = 41.8  
reistijdenmatrix[10,14] = 77.1  
reistijdenmatrix[10,15] = 66.6  
reistijdenmatrix[10,16] = 84.7  
reistijdenmatrix[10,17] = 66.2  
reistijdenmatrix[10,18] = 96.1  
reistijdenmatrix[10,19] = 110  
reistijdenmatrix[10,20] = 98  
reistijdenmatrix[10,21] = 96.8  
reistijdenmatrix[10,22] = 87.4  
reistijdenmatrix[10,23] = 89.9  
reistijdenmatrix[10,24] = 64.6  
reistijdenmatrix[10,25] = 98.8  
reistijdenmatrix[10,26] = 103.8  
reistijdenmatrix[10,27] = 110.5  
reistijdenmatrix[10,28] = 93.8  
reistijdenmatrix[10,29] = 116.8  
reistijdenmatrix[10,30] = 91.1  
reistijdenmatrix[10,31] = 196.3  
reistijdenmatrix[10,32] = 156.4  
reistijdenmatrix[10,33] = 118.6  
reistijdenmatrix[10,34] = 111.6  
reistijdenmatrix[10,35] = 80.4  
reistijdenmatrix[10,36] = 116.8  
reistijdenmatrix[10,37] = 100.7  
reistijdenmatrix[10,38] = 130.2  
reistijdenmatrix[10,39] = 158.9  
reistijdenmatrix[10,40] = 57.2  
reistijdenmatrix[11,1] = 93.6  
reistijdenmatrix[11,2] = 102.6  
reistijdenmatrix[11,3] = 92  
reistijdenmatrix[11,4] = 89.6  
reistijdenmatrix[11,5] = 88.9  
reistijdenmatrix[11,6] = 72.8  
reistijdenmatrix[11,7] = 67.6  
reistijdenmatrix[11,8] = 64.1  
reistijdenmatrix[11,9] = 52.5  
reistijdenmatrix[11,10] = 40  
reistijdenmatrix[11,11] = 12  
reistijdenmatrix[11,12] = 40.7  
reistijdenmatrix[11,13] = 25.4  
reistijdenmatrix[11,14] = 46.6

reistijdenmatrix[11,15] = 50.2  
reistijdenmatrix[11,16] = 70  
reistijdenmatrix[11,17] = 62.9  
reistijdenmatrix[11,18] = 107.5  
reistijdenmatrix[11,19] = 105.9  
reistijdenmatrix[11,20] = 94.1  
reistijdenmatrix[11,21] = 92.9  
reistijdenmatrix[11,22] = 83.4  
reistijdenmatrix[11,23] = 85.9  
reistijdenmatrix[11,24] = 60.7  
reistijdenmatrix[11,25] = 94.8  
reistijdenmatrix[11,26] = 100.1  
reistijdenmatrix[11,27] = 106.7  
reistijdenmatrix[11,28] = 90.1  
reistijdenmatrix[11,29] = 113  
reistijdenmatrix[11,30] = 87.3  
reistijdenmatrix[11,31] = 192.5  
reistijdenmatrix[11,32] = 152.6  
reistijdenmatrix[11,33] = 114.8  
reistijdenmatrix[11,34] = 95.2  
reistijdenmatrix[11,35] = 64  
reistijdenmatrix[11,36] = 100.5  
reistijdenmatrix[11,37] = 84.3  
reistijdenmatrix[11,38] = 113.8  
reistijdenmatrix[11,39] = 142.5  
reistijdenmatrix[11,40] = 70.3  
reistijdenmatrix[12,1] = 91.1  
reistijdenmatrix[12,2] = 109.1  
reistijdenmatrix[12,3] = 98.5  
reistijdenmatrix[12,4] = 106.1  
reistijdenmatrix[12,5] = 105.3  
reistijdenmatrix[12,6] = 89.3  
reistijdenmatrix[12,7] = 74.1  
reistijdenmatrix[12,8] = 53.4  
reistijdenmatrix[12,9] = 59  
reistijdenmatrix[12,10] = 49.9  
reistijdenmatrix[12,11] = 40.7  
reistijdenmatrix[12,12] = 18  
reistijdenmatrix[12,13] = 44  
reistijdenmatrix[12,14] = 45.8  
reistijdenmatrix[12,15] = 68.8  
reistijdenmatrix[12,16] = 88.6  
reistijdenmatrix[12,17] = 81.5  
reistijdenmatrix[12,18] = 126.1  
reistijdenmatrix[12,19] = 124.6  
reistijdenmatrix[12,20] = 112.7  
reistijdenmatrix[12,21] = 111.5  
reistijdenmatrix[12,22] = 102  
reistijdenmatrix[12,23] = 104.5  
reistijdenmatrix[12,24] = 79.3  
reistijdenmatrix[12,25] = 113.4  
reistijdenmatrix[12,26] = 118.7  
reistijdenmatrix[12,27] = 125.4  
reistijdenmatrix[12,28] = 108.7  
reistijdenmatrix[12,29] = 131.6

reistijdenmatrix[12,30] = 105.9  
reistijdenmatrix[12,31] = 211.1  
reistijdenmatrix[12,32] = 171.2  
reistijdenmatrix[12,33] = 133.4  
reistijdenmatrix[12,34] = 113.8  
reistijdenmatrix[12,35] = 82.7  
reistijdenmatrix[12,36] = 119.1  
reistijdenmatrix[12,37] = 102.9  
reistijdenmatrix[12,38] = 132.4  
reistijdenmatrix[12,39] = 161.1  
reistijdenmatrix[12,40] = 89  
reistijdenmatrix[13,1] = 102.9  
reistijdenmatrix[13,2] = 111.8  
reistijdenmatrix[13,3] = 101.3  
reistijdenmatrix[13,4] = 91.3  
reistijdenmatrix[13,5] = 90.3  
reistijdenmatrix[13,6] = 74.5  
reistijdenmatrix[13,7] = 76.9  
reistijdenmatrix[13,8] = 79  
reistijdenmatrix[13,9] = 61.8  
reistijdenmatrix[13,10] = 41.8  
reistijdenmatrix[13,11] = 25.4  
reistijdenmatrix[13,12] = 44  
reistijdenmatrix[13,13] = 21  
reistijdenmatrix[13,14] = 49.8  
reistijdenmatrix[13,15] = 36.6  
reistijdenmatrix[13,16] = 56.4  
reistijdenmatrix[13,17] = 43.7  
reistijdenmatrix[13,18] = 88.3  
reistijdenmatrix[13,19] = 86.7  
reistijdenmatrix[13,20] = 74.9  
reistijdenmatrix[13,21] = 73.7  
reistijdenmatrix[13,22] = 64.2  
reistijdenmatrix[13,23] = 66.7  
reistijdenmatrix[13,24] = 41.4  
reistijdenmatrix[13,25] = 75.6  
reistijdenmatrix[13,26] = 80.9  
reistijdenmatrix[13,27] = 87.5  
reistijdenmatrix[13,28] = 70.8  
reistijdenmatrix[13,29] = 93.8  
reistijdenmatrix[13,30] = 68.1  
reistijdenmatrix[13,31] = 173.3  
reistijdenmatrix[13,32] = 133.4  
reistijdenmatrix[13,33] = 95.6  
reistijdenmatrix[13,34] = 81.6  
reistijdenmatrix[13,35] = 50.5  
reistijdenmatrix[13,36] = 86.9  
reistijdenmatrix[13,37] = 70.7  
reistijdenmatrix[13,38] = 100.2  
reistijdenmatrix[13,39] = 128.9  
reistijdenmatrix[13,40] = 51.1  
reistijdenmatrix[14,1] = 128.4  
reistijdenmatrix[14,2] = 137.3  
reistijdenmatrix[14,3] = 126.8  
reistijdenmatrix[14,4] = 126.7

reistijdenmatrix[14,5] = 125.7  
reistijdenmatrix[14,6] = 109.9  
reistijdenmatrix[14,7] = 102.4  
reistijdenmatrix[14,8] = 95.4  
reistijdenmatrix[14,9] = 87.3  
reistijdenmatrix[14,10] = 77.1  
reistijdenmatrix[14,11] = 46.6  
reistijdenmatrix[14,12] = 45.8  
reistijdenmatrix[14,13] = 49.8  
reistijdenmatrix[14,14] = 14  
reistijdenmatrix[14,15] = 56.7  
reistijdenmatrix[14,16] = 82.5  
reistijdenmatrix[14,17] = 85.9  
reistijdenmatrix[14,18] = 131.9  
reistijdenmatrix[14,19] = 130.4  
reistijdenmatrix[14,20] = 118.6  
reistijdenmatrix[14,21] = 117.4  
reistijdenmatrix[14,22] = 107.9  
reistijdenmatrix[14,23] = 110.3  
reistijdenmatrix[14,24] = 85.1  
reistijdenmatrix[14,25] = 119.3  
reistijdenmatrix[14,26] = 115.5  
reistijdenmatrix[14,27] = 122.2  
reistijdenmatrix[14,28] = 105.2  
reistijdenmatrix[14,29] = 126.7  
reistijdenmatrix[14,30] = 103.1  
reistijdenmatrix[14,31] = 205.1  
reistijdenmatrix[14,32] = 165.2  
reistijdenmatrix[14,33] = 128.6  
reistijdenmatrix[14,34] = 107.9  
reistijdenmatrix[14,35] = 76.8  
reistijdenmatrix[14,36] = 113.2  
reistijdenmatrix[14,37] = 97  
reistijdenmatrix[14,38] = 126.5  
reistijdenmatrix[14,39] = 155.2  
reistijdenmatrix[14,40] = 94.8  
reistijdenmatrix[15,1] = 127.8  
reistijdenmatrix[15,2] = 136.7  
reistijdenmatrix[15,3] = 126.2  
reistijdenmatrix[15,4] = 116.2  
reistijdenmatrix[15,5] = 115.2  
reistijdenmatrix[15,6] = 99.4  
reistijdenmatrix[15,7] = 101.8  
reistijdenmatrix[15,8] = 103.9  
reistijdenmatrix[15,9] = 86.7  
reistijdenmatrix[15,10] = 66.6  
reistijdenmatrix[15,11] = 50.2  
reistijdenmatrix[15,12] = 68.8  
reistijdenmatrix[15,13] = 36.6  
reistijdenmatrix[15,14] = 56.7  
reistijdenmatrix[15,15] = 14  
reistijdenmatrix[15,16] = 35.7  
reistijdenmatrix[15,17] = 57.4  
reistijdenmatrix[15,18] = 109.5  
reistijdenmatrix[15,19] = 103



reistijdenmatrix[15,20] = 88.7  
reistijdenmatrix[15,21] = 87.3  
reistijdenmatrix[15,22] = 85.5  
reistijdenmatrix[15,23] = 81.7  
reistijdenmatrix[15,24] = 65.3  
reistijdenmatrix[15,25] = 89.8  
reistijdenmatrix[15,26] = 84.8  
reistijdenmatrix[15,27] = 86.7  
reistijdenmatrix[15,28] = 74.8  
reistijdenmatrix[15,29] = 79.8  
reistijdenmatrix[15,30] = 58.1  
reistijdenmatrix[15,31] = 158.3  
reistijdenmatrix[15,32] = 118.4  
reistijdenmatrix[15,33] = 81.8  
reistijdenmatrix[15,34] = 62.3  
reistijdenmatrix[15,35] = 31.1  
reistijdenmatrix[15,36] = 67.5  
reistijdenmatrix[15,37] = 51.4  
reistijdenmatrix[15,38] = 80.9  
reistijdenmatrix[15,39] = 109.5  
reistijdenmatrix[15,40] = 85  
reistijdenmatrix[16,1] = 145.9  
reistijdenmatrix[16,2] = 154.8  
reistijdenmatrix[16,3] = 144.3  
reistijdenmatrix[16,4] = 122.9  
reistijdenmatrix[16,5] = 110  
reistijdenmatrix[16,6] = 113.5  
reistijdenmatrix[16,7] = 119.9  
reistijdenmatrix[16,8] = 122  
reistijdenmatrix[16,9] = 104.8  
reistijdenmatrix[16,10] = 84.7  
reistijdenmatrix[16,11] = 70  
reistijdenmatrix[16,12] = 88.6  
reistijdenmatrix[16,13] = 56.4  
reistijdenmatrix[16,14] = 82.5  
reistijdenmatrix[16,15] = 35.7  
reistijdenmatrix[16,16] = 12  
reistijdenmatrix[16,17] = 32.3  
reistijdenmatrix[16,18] = 83.7  
reistijdenmatrix[16,19] = 77.2  
reistijdenmatrix[16,20] = 62.8  
reistijdenmatrix[16,21] = 61.4  
reistijdenmatrix[16,22] = 59.6  
reistijdenmatrix[16,23] = 55.9  
reistijdenmatrix[16,24] = 43.3  
reistijdenmatrix[16,25] = 63.9  
reistijdenmatrix[16,26] = 59  
reistijdenmatrix[16,27] = 61.6  
reistijdenmatrix[16,28] = 49  
reistijdenmatrix[16,29] = 54.8  
reistijdenmatrix[16,30] = 32.9  
reistijdenmatrix[16,31] = 133.3  
reistijdenmatrix[16,32] = 93.4  
reistijdenmatrix[16,33] = 56.8  
reistijdenmatrix[16,34] = 39.5

reistijdenmatrix[16,35] = 33.2  
reistijdenmatrix[16,36] = 47.5  
reistijdenmatrix[16,37] = 67.4  
reistijdenmatrix[16,38] = 66.2  
reistijdenmatrix[16,39] = 89.9  
reistijdenmatrix[16,40] = 62.9  
reistijdenmatrix[17,1] = 127.2  
reistijdenmatrix[17,2] = 136.1  
reistijdenmatrix[17,3] = 125.6  
reistijdenmatrix[17,4] = 103.8  
reistijdenmatrix[17,5] = 91  
reistijdenmatrix[17,6] = 94.4  
reistijdenmatrix[17,7] = 101.2  
reistijdenmatrix[17,8] = 103.6  
reistijdenmatrix[17,9] = 86.1  
reistijdenmatrix[17,10] = 66.2  
reistijdenmatrix[17,11] = 62.9  
reistijdenmatrix[17,12] = 81.5  
reistijdenmatrix[17,13] = 43.7  
reistijdenmatrix[17,14] = 85.9  
reistijdenmatrix[17,15] = 57.4  
reistijdenmatrix[17,16] = 32.3  
reistijdenmatrix[17,17] = 18  
reistijdenmatrix[17,18] = 71.5  
reistijdenmatrix[17,19] = 69.3  
reistijdenmatrix[17,20] = 55  
reistijdenmatrix[17,21] = 53.6  
reistijdenmatrix[17,22] = 47.5  
reistijdenmatrix[17,23] = 48  
reistijdenmatrix[17,24] = 24.3  
reistijdenmatrix[17,25] = 56.1  
reistijdenmatrix[17,26] = 51.4  
reistijdenmatrix[17,27] = 58.1  
reistijdenmatrix[17,28] = 41  
reistijdenmatrix[17,29] = 64.2  
reistijdenmatrix[17,30] = 38.7  
reistijdenmatrix[17,31] = 143.8  
reistijdenmatrix[17,32] = 103.9  
reistijdenmatrix[17,33] = 66.1  
reistijdenmatrix[17,34] = 59.2  
reistijdenmatrix[17,35] = 56.9  
reistijdenmatrix[17,36] = 71.2  
reistijdenmatrix[17,37] = 88.1  
reistijdenmatrix[17,38] = 89.9  
reistijdenmatrix[17,39] = 113.6  
reistijdenmatrix[17,40] = 43.8  
reistijdenmatrix[18,1] = 121.9  
reistijdenmatrix[18,2] = 113.9  
reistijdenmatrix[18,3] = 103.2  
reistijdenmatrix[18,4] = 57  
reistijdenmatrix[18,5] = 46.7  
reistijdenmatrix[18,6] = 68.1  
reistijdenmatrix[18,7] = 110.1  
reistijdenmatrix[18,8] = 121.3  
reistijdenmatrix[18,9] = 100.4

reistijdenmatrix[18,10] = 96.1  
reistijdenmatrix[18,11] = 107.5  
reistijdenmatrix[18,12] = 126.1  
reistijdenmatrix[18,13] = 88.3  
reistijdenmatrix[18,14] = 131.9  
reistijdenmatrix[18,15] = 109.5  
reistijdenmatrix[18,16] = 83.7  
reistijdenmatrix[18,17] = 71.5  
reistijdenmatrix[18,18] = 15  
reistijdenmatrix[18,19] = 26.3  
reistijdenmatrix[18,20] = 42.2  
reistijdenmatrix[18,21] = 50  
reistijdenmatrix[18,22] = 32.8  
reistijdenmatrix[18,23] = 43.7  
reistijdenmatrix[18,24] = 55.7  
reistijdenmatrix[18,25] = 58.6  
reistijdenmatrix[18,26] = 72.4  
reistijdenmatrix[18,27] = 79.8  
reistijdenmatrix[18,28] = 71.4  
reistijdenmatrix[18,29] = 96.2  
reistijdenmatrix[18,30] = 90.6  
reistijdenmatrix[18,31] = 192.6  
reistijdenmatrix[18,32] = 152.7  
reistijdenmatrix[18,33] = 117.2  
reistijdenmatrix[18,34] = 111.2  
reistijdenmatrix[18,35] = 108.2  
reistijdenmatrix[18,36] = 122.5  
reistijdenmatrix[18,37] = 141.3  
reistijdenmatrix[18,38] = 141.2  
reistijdenmatrix[18,39] = 164.9  
reistijdenmatrix[18,40] = 61  
reistijdenmatrix[19,1] = 144.7  
reistijdenmatrix[19,2] = 136.7  
reistijdenmatrix[19,3] = 126  
reistijdenmatrix[19,4] = 79.8  
reistijdenmatrix[19,5] = 69.5  
reistijdenmatrix[19,6] = 90.9  
reistijdenmatrix[19,7] = 132.9  
reistijdenmatrix[19,8] = 144.1  
reistijdenmatrix[19,9] = 123.2  
reistijdenmatrix[19,10] = 110  
reistijdenmatrix[19,11] = 105.9  
reistijdenmatrix[19,12] = 124.6  
reistijdenmatrix[19,13] = 86.7  
reistijdenmatrix[19,14] = 130.4  
reistijdenmatrix[19,15] = 103  
reistijdenmatrix[19,16] = 77.2  
reistijdenmatrix[19,17] = 69.3  
reistijdenmatrix[19,18] = 26.3  
reistijdenmatrix[19,19] = 15  
reistijdenmatrix[19,20] = 22.5  
reistijdenmatrix[19,21] = 30.3  
reistijdenmatrix[19,22] = 24.8  
reistijdenmatrix[19,23] = 41.6  
reistijdenmatrix[19,24] = 54.2

reistijdenmatrix[19,25] = 48  
reistijdenmatrix[19,26] = 62  
reistijdenmatrix[19,27] = 70.1  
reistijdenmatrix[19,28] = 61.7  
reistijdenmatrix[19,29] = 86.6  
reistijdenmatrix[19,30] = 83.5  
reistijdenmatrix[19,31] = 182.9  
reistijdenmatrix[19,32] = 143.1  
reistijdenmatrix[19,33] = 107.5  
reistijdenmatrix[19,34] = 104.6  
reistijdenmatrix[19,35] = 101.5  
reistijdenmatrix[19,36] = 115.9  
reistijdenmatrix[19,37] = 134.7  
reistijdenmatrix[19,38] = 134.6  
reistijdenmatrix[19,39] = 158.3  
reistijdenmatrix[19,40] = 67.6  
reistijdenmatrix[20,1] = 159  
reistijdenmatrix[20,2] = 152.4  
reistijdenmatrix[20,3] = 141.8  
reistijdenmatrix[20,4] = 95.6  
reistijdenmatrix[20,5] = 85.3  
reistijdenmatrix[20,6] = 106.4  
reistijdenmatrix[20,7] = 133  
reistijdenmatrix[20,8] = 135.3  
reistijdenmatrix[20,9] = 117.9  
reistijdenmatrix[20,10] = 98  
reistijdenmatrix[20,11] = 94.1  
reistijdenmatrix[20,12] = 112.7  
reistijdenmatrix[20,13] = 74.9  
reistijdenmatrix[20,14] = 118.6  
reistijdenmatrix[20,15] = 88.7  
reistijdenmatrix[20,16] = 62.8  
reistijdenmatrix[20,17] = 55  
reistijdenmatrix[20,18] = 42.2  
reistijdenmatrix[20,19] = 22.5  
reistijdenmatrix[20,20] = 8  
reistijdenmatrix[20,21] = 14.2  
reistijdenmatrix[20,22] = 14.9  
reistijdenmatrix[20,23] = 27.3  
reistijdenmatrix[20,24] = 41.8  
reistijdenmatrix[20,25] = 33.7  
reistijdenmatrix[20,26] = 47.6  
reistijdenmatrix[20,27] = 55.8  
reistijdenmatrix[20,28] = 47.4  
reistijdenmatrix[20,29] = 72.2  
reistijdenmatrix[20,30] = 69.1  
reistijdenmatrix[20,31] = 168.6  
reistijdenmatrix[20,32] = 128.7  
reistijdenmatrix[20,33] = 93.1  
reistijdenmatrix[20,34] = 90.3  
reistijdenmatrix[20,35] = 87.2  
reistijdenmatrix[20,36] = 101.5  
reistijdenmatrix[20,37] = 120.4  
reistijdenmatrix[20,38] = 120.2  
reistijdenmatrix[20,39] = 143.9

reistijdenmatrix[20,40] = 55.8  
reistijdenmatrix[21,1] = 157.8  
reistijdenmatrix[21,2] = 151.2  
reistijdenmatrix[21,3] = 140.6  
reistijdenmatrix[21,4] = 103.4  
reistijdenmatrix[21,5] = 93.2  
reistijdenmatrix[21,6] = 105.2  
reistijdenmatrix[21,7] = 131.8  
reistijdenmatrix[21,8] = 134.1  
reistijdenmatrix[21,9] = 116.7  
reistijdenmatrix[21,10] = 96.8  
reistijdenmatrix[21,11] = 92.9  
reistijdenmatrix[21,12] = 111.5  
reistijdenmatrix[21,13] = 73.7  
reistijdenmatrix[21,14] = 117.4  
reistijdenmatrix[21,15] = 87.3  
reistijdenmatrix[21,16] = 61.4  
reistijdenmatrix[21,17] = 53.6  
reistijdenmatrix[21,18] = 50  
reistijdenmatrix[21,19] = 30.3  
reistijdenmatrix[21,20] = 14.2  
reistijdenmatrix[21,21] = 8  
reistijdenmatrix[21,22] = 24.8  
reistijdenmatrix[21,23] = 26.6  
reistijdenmatrix[21,24] = 40.6  
reistijdenmatrix[21,25] = 27.1  
reistijdenmatrix[21,26] = 45.9  
reistijdenmatrix[21,27] = 54.1  
reistijdenmatrix[21,28] = 45  
reistijdenmatrix[21,29] = 70.6  
reistijdenmatrix[21,30] = 67.7  
reistijdenmatrix[21,31] = 167  
reistijdenmatrix[21,32] = 127.1  
reistijdenmatrix[21,33] = 91.5  
reistijdenmatrix[21,34] = 88.9  
reistijdenmatrix[21,35] = 85.8  
reistijdenmatrix[21,36] = 100.1  
reistijdenmatrix[21,37] = 119  
reistijdenmatrix[21,38] = 118.8  
reistijdenmatrix[21,39] = 142.5  
reistijdenmatrix[21,40] = 54.6  
reistijdenmatrix[22,1] = 148.4  
reistijdenmatrix[22,2] = 141.7  
reistijdenmatrix[22,3] = 131.1  
reistijdenmatrix[22,4] = 86.1  
reistijdenmatrix[22,5] = 75.9  
reistijdenmatrix[22,6] = 95.7  
reistijdenmatrix[22,7] = 122.4  
reistijdenmatrix[22,8] = 124.8  
reistijdenmatrix[22,9] = 107.3  
reistijdenmatrix[22,10] = 87.4  
reistijdenmatrix[22,11] = 83.4  
reistijdenmatrix[22,12] = 102  
reistijdenmatrix[22,13] = 64.2  
reistijdenmatrix[22,14] = 107.9

reistijdenmatrix[22,15] = 85.5  
reistijdenmatrix[22,16] = 59.6  
reistijdenmatrix[22,17] = 47.5  
reistijdenmatrix[22,18] = 32.8  
reistijdenmatrix[22,19] = 24.8  
reistijdenmatrix[22,20] = 14.9  
reistijdenmatrix[22,21] = 24.8  
reistijdenmatrix[22,22] = 7  
reistijdenmatrix[22,23] = 19.4  
reistijdenmatrix[22,24] = 31.6  
reistijdenmatrix[22,25] = 34.4  
reistijdenmatrix[22,26] = 48.1  
reistijdenmatrix[22,27] = 55.6  
reistijdenmatrix[22,28] = 47.1  
reistijdenmatrix[22,29] = 72  
reistijdenmatrix[22,30] = 66.3  
reistijdenmatrix[22,31] = 168.3  
reistijdenmatrix[22,32] = 128.5  
reistijdenmatrix[22,33] = 92.9  
reistijdenmatrix[22,34] = 87.2  
reistijdenmatrix[22,35] = 84.2  
reistijdenmatrix[22,36] = 98.5  
reistijdenmatrix[22,37] = 117.2  
reistijdenmatrix[22,38] = 117.2  
reistijdenmatrix[22,39] = 140.9  
reistijdenmatrix[22,40] = 45.1  
reistijdenmatrix[23,1] = 150.9  
reistijdenmatrix[23,2] = 144.2  
reistijdenmatrix[23,3] = 133.6  
reistijdenmatrix[23,4] = 97  
reistijdenmatrix[23,5] = 86.7  
reistijdenmatrix[23,6] = 98.2  
reistijdenmatrix[23,7] = 124.9  
reistijdenmatrix[23,8] = 127.2  
reistijdenmatrix[23,9] = 109.8  
reistijdenmatrix[23,10] = 89.9  
reistijdenmatrix[23,11] = 85.9  
reistijdenmatrix[23,12] = 104.5  
reistijdenmatrix[23,13] = 66.7  
reistijdenmatrix[23,14] = 110.3  
reistijdenmatrix[23,15] = 81.7  
reistijdenmatrix[23,16] = 55.9  
reistijdenmatrix[23,17] = 48  
reistijdenmatrix[23,18] = 43.7  
reistijdenmatrix[23,19] = 41.6  
reistijdenmatrix[23,20] = 27.3  
reistijdenmatrix[23,21] = 26.6  
reistijdenmatrix[23,22] = 19.4  
reistijdenmatrix[23,23] = 23  
reistijdenmatrix[23,24] = 33.9  
reistijdenmatrix[23,25] = 30.5  
reistijdenmatrix[23,26] = 44.3  
reistijdenmatrix[23,27] = 51.7  
reistijdenmatrix[23,28] = 43.3  
reistijdenmatrix[23,29] = 68.1

reistijdenmatrix[23,30] = 62.4  
reistijdenmatrix[23,31] = 164.5  
reistijdenmatrix[23,32] = 124.6  
reistijdenmatrix[23,33] = 89  
reistijdenmatrix[23,34] = 83.4  
reistijdenmatrix[23,35] = 80.4  
reistijdenmatrix[23,36] = 94.7  
reistijdenmatrix[23,37] = 113.5  
reistijdenmatrix[23,38] = 113.4  
reistijdenmatrix[23,39] = 137.2  
reistijdenmatrix[23,40] = 47.6  
reistijdenmatrix[24,1] = 125.6  
reistijdenmatrix[24,2] = 128.9  
reistijdenmatrix[24,3] = 118.3  
reistijdenmatrix[24,4] = 92.3  
reistijdenmatrix[24,5] = 79.4  
reistijdenmatrix[24,6] = 82.9  
reistijdenmatrix[24,7] = 99.6  
reistijdenmatrix[24,8] = 101.9  
reistijdenmatrix[24,9] = 84.5  
reistijdenmatrix[24,10] = 64.6  
reistijdenmatrix[24,11] = 60.7  
reistijdenmatrix[24,12] = 79.3  
reistijdenmatrix[24,13] = 41.4  
reistijdenmatrix[24,14] = 85.1  
reistijdenmatrix[24,15] = 65.3  
reistijdenmatrix[24,16] = 43.3  
reistijdenmatrix[24,17] = 24.3  
reistijdenmatrix[24,18] = 55.7  
reistijdenmatrix[24,19] = 54.2  
reistijdenmatrix[24,20] = 41.8  
reistijdenmatrix[24,21] = 40.6  
reistijdenmatrix[24,22] = 31.6  
reistijdenmatrix[24,23] = 33.9  
reistijdenmatrix[24,24] = 10  
reistijdenmatrix[24,25] = 42.7  
reistijdenmatrix[24,26] = 56.5  
reistijdenmatrix[24,27] = 63.9  
reistijdenmatrix[24,28] = 51.5  
reistijdenmatrix[24,29] = 74.7  
reistijdenmatrix[24,30] = 49.2  
reistijdenmatrix[24,31] = 154.3  
reistijdenmatrix[24,32] = 114.4  
reistijdenmatrix[24,33] = 77.2  
reistijdenmatrix[24,34] = 70.3  
reistijdenmatrix[24,35] = 67.8  
reistijdenmatrix[24,36] = 82.2  
reistijdenmatrix[24,37] = 96  
reistijdenmatrix[24,38] = 100.9  
reistijdenmatrix[24,39] = 124.6  
reistijdenmatrix[24,40] = 32.1  
reistijdenmatrix[25,1] = 159.8  
reistijdenmatrix[25,2] = 153.1  
reistijdenmatrix[25,3] = 142.5  
reistijdenmatrix[25,4] = 111.9

reistijdenmatrix[25,5] = 101.7  
reistijdenmatrix[25,6] = 107.1  
reistijdenmatrix[25,7] = 133.8  
reistijdenmatrix[25,8] = 136.2  
reistijdenmatrix[25,9] = 118.7  
reistijdenmatrix[25,10] = 98.8  
reistijdenmatrix[25,11] = 94.8  
reistijdenmatrix[25,12] = 113.4  
reistijdenmatrix[25,13] = 75.6  
reistijdenmatrix[25,14] = 119.3  
reistijdenmatrix[25,15] = 89.8  
reistijdenmatrix[25,16] = 63.9  
reistijdenmatrix[25,17] = 56.1  
reistijdenmatrix[25,18] = 58.6  
reistijdenmatrix[25,19] = 48  
reistijdenmatrix[25,20] = 33.7  
reistijdenmatrix[25,21] = 27.1  
reistijdenmatrix[25,22] = 34.4  
reistijdenmatrix[25,23] = 30.5  
reistijdenmatrix[25,24] = 42.7  
reistijdenmatrix[25,25] = 11  
reistijdenmatrix[25,26] = 25.5  
reistijdenmatrix[25,27] = 39  
reistijdenmatrix[25,28] = 30.8  
reistijdenmatrix[25,29] = 55.7  
reistijdenmatrix[25,30] = 67.6  
reistijdenmatrix[25,31] = 151.9  
reistijdenmatrix[25,32] = 112.1  
reistijdenmatrix[25,33] = 76.7  
reistijdenmatrix[25,34] = 90.6  
reistijdenmatrix[25,35] = 88.2  
reistijdenmatrix[25,36] = 102.6  
reistijdenmatrix[25,37] = 121.4  
reistijdenmatrix[25,38] = 121.3  
reistijdenmatrix[25,39] = 145  
reistijdenmatrix[25,40] = 56.5  
reistijdenmatrix[26,1] = 165  
reistijdenmatrix[26,2] = 167.1  
reistijdenmatrix[26,3] = 156.5  
reistijdenmatrix[26,4] = 125.7  
reistijdenmatrix[26,5] = 115.4  
reistijdenmatrix[26,6] = 121.1  
reistijdenmatrix[26,7] = 139  
reistijdenmatrix[26,8] = 141.1  
reistijdenmatrix[26,9] = 123.9  
reistijdenmatrix[26,10] = 103.8  
reistijdenmatrix[26,11] = 100.1  
reistijdenmatrix[26,12] = 118.7  
reistijdenmatrix[26,13] = 80.9  
reistijdenmatrix[26,14] = 115.5  
reistijdenmatrix[26,15] = 84.8  
reistijdenmatrix[26,16] = 59  
reistijdenmatrix[26,17] = 51.4  
reistijdenmatrix[26,18] = 72.4  
reistijdenmatrix[26,19] = 62



reistijdenmatrix[26,20] = 47.6  
reistijdenmatrix[26,21] = 45.9  
reistijdenmatrix[26,22] = 48.1  
reistijdenmatrix[26,23] = 44.3  
reistijdenmatrix[26,24] = 56.5  
reistijdenmatrix[26,25] = 25.5  
reistijdenmatrix[26,26] = 12  
reistijdenmatrix[26,27] = 21.4  
reistijdenmatrix[26,28] = 28.3  
reistijdenmatrix[26,29] = 37.8  
reistijdenmatrix[26,30] = 49.7  
reistijdenmatrix[26,31] = 134.2  
reistijdenmatrix[26,32] = 94.3  
reistijdenmatrix[26,33] = 58.8  
reistijdenmatrix[26,34] = 72.7  
reistijdenmatrix[26,35] = 83.4  
reistijdenmatrix[26,36] = 95.9  
reistijdenmatrix[26,37] = 116.6  
reistijdenmatrix[26,38] = 114.4  
reistijdenmatrix[26,39] = 138.2  
reistijdenmatrix[26,40] = 70.5  
reistijdenmatrix[27,1] = 171.7  
reistijdenmatrix[27,2] = 174.6  
reistijdenmatrix[27,3] = 164  
reistijdenmatrix[27,4] = 133.1  
reistijdenmatrix[27,5] = 122.9  
reistijdenmatrix[27,6] = 128.5  
reistijdenmatrix[27,7] = 145.7  
reistijdenmatrix[27,8] = 147.7  
reistijdenmatrix[27,9] = 130.6  
reistijdenmatrix[27,10] = 110.5  
reistijdenmatrix[27,11] = 106.7  
reistijdenmatrix[27,12] = 125.4  
reistijdenmatrix[27,13] = 87.5  
reistijdenmatrix[27,14] = 122.2  
reistijdenmatrix[27,15] = 86.7  
reistijdenmatrix[27,16] = 61.6  
reistijdenmatrix[27,17] = 58.1  
reistijdenmatrix[27,18] = 79.8  
reistijdenmatrix[27,19] = 70.1  
reistijdenmatrix[27,20] = 55.8  
reistijdenmatrix[27,21] = 54.1  
reistijdenmatrix[27,22] = 55.6  
reistijdenmatrix[27,23] = 51.7  
reistijdenmatrix[27,24] = 63.9  
reistijdenmatrix[27,25] = 39  
reistijdenmatrix[27,26] = 21.4  
reistijdenmatrix[27,27] = 12  
reistijdenmatrix[27,28] = 35.8  
reistijdenmatrix[27,29] = 29.2  
reistijdenmatrix[27,30] = 48.4  
reistijdenmatrix[27,31] = 132.7  
reistijdenmatrix[27,32] = 91.8  
reistijdenmatrix[27,33] = 57.5  
reistijdenmatrix[27,34] = 71.4

reistijdenmatrix[27,35] = 86  
reistijdenmatrix[27,36] = 94.5  
reistijdenmatrix[27,37] = 118.4  
reistijdenmatrix[27,38] = 113  
reistijdenmatrix[27,39] = 136.7  
reistijdenmatrix[27,40] = 77.9  
reistijdenmatrix[28,1] = 154.8  
reistijdenmatrix[28,2] = 163.7  
reistijdenmatrix[28,3] = 153.2  
reistijdenmatrix[28,4] = 124.7  
reistijdenmatrix[28,5] = 114.5  
reistijdenmatrix[28,6] = 120.1  
reistijdenmatrix[28,7] = 128.8  
reistijdenmatrix[28,8] = 131  
reistijdenmatrix[28,9] = 113.7  
reistijdenmatrix[28,10] = 93.8  
reistijdenmatrix[28,11] = 90.1  
reistijdenmatrix[28,12] = 108.7  
reistijdenmatrix[28,13] = 70.8  
reistijdenmatrix[28,14] = 105.2  
reistijdenmatrix[28,15] = 74.8  
reistijdenmatrix[28,16] = 49  
reistijdenmatrix[28,17] = 41  
reistijdenmatrix[28,18] = 71.4  
reistijdenmatrix[28,19] = 61.7  
reistijdenmatrix[28,20] = 47.4  
reistijdenmatrix[28,21] = 45  
reistijdenmatrix[28,22] = 47.1  
reistijdenmatrix[28,23] = 43.3  
reistijdenmatrix[28,24] = 51.5  
reistijdenmatrix[28,25] = 30.8  
reistijdenmatrix[28,26] = 28.3  
reistijdenmatrix[28,27] = 35.8  
reistijdenmatrix[28,28] = 13  
reistijdenmatrix[28,29] = 44.3  
reistijdenmatrix[28,30] = 47.8  
reistijdenmatrix[28,31] = 132.4  
reistijdenmatrix[28,32] = 92.5  
reistijdenmatrix[28,33] = 57.2  
reistijdenmatrix[28,34] = 71.1  
reistijdenmatrix[28,35] = 73.4  
reistijdenmatrix[28,36] = 87.7  
reistijdenmatrix[28,37] = 106.6  
reistijdenmatrix[28,38] = 106.4  
reistijdenmatrix[28,39] = 130.1  
reistijdenmatrix[28,40] = 69.5  
reistijdenmatrix[29,1] = 177.9  
reistijdenmatrix[29,2] = 186.9  
reistijdenmatrix[29,3] = 176.3  
reistijdenmatrix[29,4] = 149.5  
reistijdenmatrix[29,5] = 139.3  
reistijdenmatrix[29,6] = 145  
reistijdenmatrix[29,7] = 151.9  
reistijdenmatrix[29,8] = 154  
reistijdenmatrix[29,9] = 136.8

reistijdenmatrix[29,10] = 116.8  
reistijdenmatrix[29,11] = 113  
reistijdenmatrix[29,12] = 131.6  
reistijdenmatrix[29,13] = 93.8  
reistijdenmatrix[29,14] = 126.7  
reistijdenmatrix[29,15] = 79.8  
reistijdenmatrix[29,16] = 54.8  
reistijdenmatrix[29,17] = 64.2  
reistijdenmatrix[29,18] = 96.2  
reistijdenmatrix[29,19] = 86.6  
reistijdenmatrix[29,20] = 72.2  
reistijdenmatrix[29,21] = 70.6  
reistijdenmatrix[29,22] = 72  
reistijdenmatrix[29,23] = 68.1  
reistijdenmatrix[29,24] = 74.7  
reistijdenmatrix[29,25] = 55.7  
reistijdenmatrix[29,26] = 37.8  
reistijdenmatrix[29,27] = 29.2  
reistijdenmatrix[29,28] = 44.3  
reistijdenmatrix[29,29] = 16  
reistijdenmatrix[29,30] = 42.1  
reistijdenmatrix[29,31] = 125.9  
reistijdenmatrix[29,32] = 84.9  
reistijdenmatrix[29,33] = 50.8  
reistijdenmatrix[29,34] = 64.7  
reistijdenmatrix[29,35] = 79.2  
reistijdenmatrix[29,36] = 87.7  
reistijdenmatrix[29,37] = 111.5  
reistijdenmatrix[29,38] = 106.2  
reistijdenmatrix[29,39] = 129.9  
reistijdenmatrix[29,40] = 94.4  
reistijdenmatrix[30,1] = 152.3  
reistijdenmatrix[30,2] = 161.2  
reistijdenmatrix[30,3] = 150.6  
reistijdenmatrix[30,4] = 129.2  
reistijdenmatrix[30,5] = 116.3  
reistijdenmatrix[30,6] = 119.8  
reistijdenmatrix[30,7] = 126.3  
reistijdenmatrix[30,8] = 128.3  
reistijdenmatrix[30,9] = 111.2  
reistijdenmatrix[30,10] = 91.1  
reistijdenmatrix[30,11] = 87.3  
reistijdenmatrix[30,12] = 105.9  
reistijdenmatrix[30,13] = 68.1  
reistijdenmatrix[30,14] = 103.1  
reistijdenmatrix[30,15] = 58.1  
reistijdenmatrix[30,16] = 32.9  
reistijdenmatrix[30,17] = 38.7  
reistijdenmatrix[30,18] = 90.6  
reistijdenmatrix[30,19] = 83.5  
reistijdenmatrix[30,20] = 69.1  
reistijdenmatrix[30,21] = 67.7  
reistijdenmatrix[30,22] = 66.3  
reistijdenmatrix[30,23] = 62.4  
reistijdenmatrix[30,24] = 49.2

reistijdenmatrix[30,25] = 67.6  
reistijdenmatrix[30,26] = 49.7  
reistijdenmatrix[30,27] = 48.4  
reistijdenmatrix[30,28] = 47.8  
reistijdenmatrix[30,29] = 42.1  
reistijdenmatrix[30,30] = 16  
reistijdenmatrix[30,31] = 120.1  
reistijdenmatrix[30,32] = 80.2  
reistijdenmatrix[30,33] = 44.5  
reistijdenmatrix[30,34] = 46.5  
reistijdenmatrix[30,35] = 57.4  
reistijdenmatrix[30,36] = 71.7  
reistijdenmatrix[30,37] = 89.8  
reistijdenmatrix[30,38] = 90.4  
reistijdenmatrix[30,39] = 114.1  
reistijdenmatrix[30,40] = 69.2  
reistijdenmatrix[31,1] = 257.4  
reistijdenmatrix[31,2] = 266.4  
reistijdenmatrix[31,3] = 255.8  
reistijdenmatrix[31,4] = 234.3  
reistijdenmatrix[31,5] = 221.4  
reistijdenmatrix[31,6] = 224.9  
reistijdenmatrix[31,7] = 231.4  
reistijdenmatrix[31,8] = 233.5  
reistijdenmatrix[31,9] = 216.3  
reistijdenmatrix[31,10] = 196.3  
reistijdenmatrix[31,11] = 192.5  
reistijdenmatrix[31,12] = 211.1  
reistijdenmatrix[31,13] = 173.3  
reistijdenmatrix[31,14] = 205.1  
reistijdenmatrix[31,15] = 158.3  
reistijdenmatrix[31,16] = 133.3  
reistijdenmatrix[31,17] = 143.8  
reistijdenmatrix[31,18] = 192.6  
reistijdenmatrix[31,19] = 182.9  
reistijdenmatrix[31,20] = 168.6  
reistijdenmatrix[31,21] = 167  
reistijdenmatrix[31,22] = 168.3  
reistijdenmatrix[31,23] = 164.5  
reistijdenmatrix[31,24] = 154.3  
reistijdenmatrix[31,25] = 151.9  
reistijdenmatrix[31,26] = 134.2  
reistijdenmatrix[31,27] = 132.7  
reistijdenmatrix[31,28] = 132.4  
reistijdenmatrix[31,29] = 125.9  
reistijdenmatrix[31,30] = 120.1  
reistijdenmatrix[31,31] = 16  
reistijdenmatrix[31,32] = 75.8  
reistijdenmatrix[31,33] = 90.9  
reistijdenmatrix[31,34] = 116.6  
reistijdenmatrix[31,35] = 145.3  
reistijdenmatrix[31,36] = 139.9  
reistijdenmatrix[31,37] = 165.6  
reistijdenmatrix[31,38] = 158.5  
reistijdenmatrix[31,39] = 182.2

reistijdenmatrix[31,40] = 174.3  
reistijdenmatrix[32,1] = 217.5  
reistijdenmatrix[32,2] = 226.5  
reistijdenmatrix[32,3] = 215.9  
reistijdenmatrix[32,4] = 194.4  
reistijdenmatrix[32,5] = 181.6  
reistijdenmatrix[32,6] = 185  
reistijdenmatrix[32,7] = 191.6  
reistijdenmatrix[32,8] = 193.6  
reistijdenmatrix[32,9] = 176.5  
reistijdenmatrix[32,10] = 156.4  
reistijdenmatrix[32,11] = 152.6  
reistijdenmatrix[32,12] = 171.2  
reistijdenmatrix[32,13] = 133.4  
reistijdenmatrix[32,14] = 165.2  
reistijdenmatrix[32,15] = 118.4  
reistijdenmatrix[32,16] = 93.4  
reistijdenmatrix[32,17] = 103.9  
reistijdenmatrix[32,18] = 152.7  
reistijdenmatrix[32,19] = 143.1  
reistijdenmatrix[32,20] = 128.7  
reistijdenmatrix[32,21] = 127.1  
reistijdenmatrix[32,22] = 128.5  
reistijdenmatrix[32,23] = 124.6  
reistijdenmatrix[32,24] = 114.4  
reistijdenmatrix[32,25] = 112.1  
reistijdenmatrix[32,26] = 94.3  
reistijdenmatrix[32,27] = 91.8  
reistijdenmatrix[32,28] = 92.5  
reistijdenmatrix[32,29] = 84.9  
reistijdenmatrix[32,30] = 80.2  
reistijdenmatrix[32,31] = 75.8  
reistijdenmatrix[32,32] = 16  
reistijdenmatrix[32,33] = 50.9  
reistijdenmatrix[32,34] = 76.6  
reistijdenmatrix[32,35] = 105.5  
reistijdenmatrix[32,36] = 99.9  
reistijdenmatrix[32,37] = 125.6  
reistijdenmatrix[32,38] = 118.5  
reistijdenmatrix[32,39] = 142.2  
reistijdenmatrix[32,40] = 134.4  
reistijdenmatrix[33,1] = 179.7  
reistijdenmatrix[33,2] = 188.7  
reistijdenmatrix[33,3] = 178.1  
reistijdenmatrix[33,4] = 156.7  
reistijdenmatrix[33,5] = 143.9  
reistijdenmatrix[33,6] = 147.3  
reistijdenmatrix[33,7] = 153.8  
reistijdenmatrix[33,8] = 155.8  
reistijdenmatrix[33,9] = 138.6  
reistijdenmatrix[33,10] = 118.6  
reistijdenmatrix[33,11] = 114.8  
reistijdenmatrix[33,12] = 133.4  
reistijdenmatrix[33,13] = 95.6  
reistijdenmatrix[33,14] = 128.6

reistijdenmatrix[33,15] = 81.8  
reistijdenmatrix[33,16] = 56.8  
reistijdenmatrix[33,17] = 66.1  
reistijdenmatrix[33,18] = 117.2  
reistijdenmatrix[33,19] = 107.5  
reistijdenmatrix[33,20] = 93.1  
reistijdenmatrix[33,21] = 91.5  
reistijdenmatrix[33,22] = 92.9  
reistijdenmatrix[33,23] = 89  
reistijdenmatrix[33,24] = 77.2  
reistijdenmatrix[33,25] = 76.7  
reistijdenmatrix[33,26] = 58.8  
reistijdenmatrix[33,27] = 57.5  
reistijdenmatrix[33,28] = 57.2  
reistijdenmatrix[33,29] = 50.8  
reistijdenmatrix[33,30] = 44.5  
reistijdenmatrix[33,31] = 90.9  
reistijdenmatrix[33,32] = 50.9  
reistijdenmatrix[33,33] = 19  
reistijdenmatrix[33,34] = 37.3  
reistijdenmatrix[33,35] = 66.9  
reistijdenmatrix[33,36] = 60.6  
reistijdenmatrix[33,37] = 86.3  
reistijdenmatrix[33,38] = 79.2  
reistijdenmatrix[33,39] = 102.9  
reistijdenmatrix[33,40] = 96.7  
reistijdenmatrix[34,1] = 172.7  
reistijdenmatrix[34,2] = 181.7  
reistijdenmatrix[34,3] = 171.1  
reistijdenmatrix[34,4] = 149.8  
reistijdenmatrix[34,5] = 137  
reistijdenmatrix[34,6] = 140.4  
reistijdenmatrix[34,7] = 146.7  
reistijdenmatrix[34,8] = 148.8  
reistijdenmatrix[34,9] = 131.6  
reistijdenmatrix[34,10] = 111.6  
reistijdenmatrix[34,11] = 95.2  
reistijdenmatrix[34,12] = 113.8  
reistijdenmatrix[34,13] = 81.6  
reistijdenmatrix[34,14] = 107.9  
reistijdenmatrix[34,15] = 62.3  
reistijdenmatrix[34,16] = 39.5  
reistijdenmatrix[34,17] = 59.2  
reistijdenmatrix[34,18] = 111.2  
reistijdenmatrix[34,19] = 104.6  
reistijdenmatrix[34,20] = 90.3  
reistijdenmatrix[34,21] = 88.9  
reistijdenmatrix[34,22] = 87.2  
reistijdenmatrix[34,23] = 83.4  
reistijdenmatrix[34,24] = 70.3  
reistijdenmatrix[34,25] = 90.6  
reistijdenmatrix[34,26] = 72.7  
reistijdenmatrix[34,27] = 71.4  
reistijdenmatrix[34,28] = 71.1  
reistijdenmatrix[34,29] = 64.7

reistijdenmatrix[34,30] = 46.5  
reistijdenmatrix[34,31] = 116.6  
reistijdenmatrix[34,32] = 76.6  
reistijdenmatrix[34,33] = 37.3  
reistijdenmatrix[34,34] = 17  
reistijdenmatrix[34,35] = 45.3  
reistijdenmatrix[34,36] = 37.2  
reistijdenmatrix[34,37] = 62.9  
reistijdenmatrix[34,38] = 55.9  
reistijdenmatrix[34,39] = 79.6  
reistijdenmatrix[34,40] = 89.8  
reistijdenmatrix[35,1] = 141.6  
reistijdenmatrix[35,2] = 150.5  
reistijdenmatrix[35,3] = 139.9  
reistijdenmatrix[35,4] = 129.9  
reistijdenmatrix[35,5] = 129  
reistijdenmatrix[35,6] = 113.1  
reistijdenmatrix[35,7] = 115.6  
reistijdenmatrix[35,8] = 117.6  
reistijdenmatrix[35,9] = 100.5  
reistijdenmatrix[35,10] = 80.4  
reistijdenmatrix[35,11] = 64  
reistijdenmatrix[35,12] = 82.7  
reistijdenmatrix[35,13] = 50.5  
reistijdenmatrix[35,14] = 76.8  
reistijdenmatrix[35,15] = 31.1  
reistijdenmatrix[35,16] = 33.2  
reistijdenmatrix[35,17] = 56.9  
reistijdenmatrix[35,18] = 108.2  
reistijdenmatrix[35,19] = 101.5  
reistijdenmatrix[35,20] = 87.2  
reistijdenmatrix[35,21] = 85.8  
reistijdenmatrix[35,22] = 84.2  
reistijdenmatrix[35,23] = 80.4  
reistijdenmatrix[35,24] = 67.8  
reistijdenmatrix[35,25] = 88.2  
reistijdenmatrix[35,26] = 83.4  
reistijdenmatrix[35,27] = 86  
reistijdenmatrix[35,28] = 73.4  
reistijdenmatrix[35,29] = 79.2  
reistijdenmatrix[35,30] = 57.4  
reistijdenmatrix[35,31] = 145.3  
reistijdenmatrix[35,32] = 105.5  
reistijdenmatrix[35,33] = 66.9  
reistijdenmatrix[35,34] = 45.3  
reistijdenmatrix[35,35] = 16  
reistijdenmatrix[35,36] = 40.4  
reistijdenmatrix[35,37] = 49.8  
reistijdenmatrix[35,38] = 61.7  
reistijdenmatrix[35,39] = 85.4  
reistijdenmatrix[35,40] = 87.4  
reistijdenmatrix[36,1] = 178  
reistijdenmatrix[36,2] = 186.9  
reistijdenmatrix[36,3] = 176.4  
reistijdenmatrix[36,4] = 161.7

reistijdenmatrix[36,5] = 148.9  
reistijdenmatrix[36,6] = 149.6  
reistijdenmatrix[36,7] = 152  
reistijdenmatrix[36,8] = 154.1  
reistijdenmatrix[36,9] = 136.9  
reistijdenmatrix[36,10] = 116.8  
reistijdenmatrix[36,11] = 100.5  
reistijdenmatrix[36,12] = 119.1  
reistijdenmatrix[36,13] = 86.9  
reistijdenmatrix[36,14] = 113.2  
reistijdenmatrix[36,15] = 67.5  
reistijdenmatrix[36,16] = 47.5  
reistijdenmatrix[36,17] = 71.2  
reistijdenmatrix[36,18] = 122.5  
reistijdenmatrix[36,19] = 115.9  
reistijdenmatrix[36,20] = 101.5  
reistijdenmatrix[36,21] = 100.1  
reistijdenmatrix[36,22] = 98.5  
reistijdenmatrix[36,23] = 94.7  
reistijdenmatrix[36,24] = 82.2  
reistijdenmatrix[36,25] = 102.6  
reistijdenmatrix[36,26] = 95.9  
reistijdenmatrix[36,27] = 94.5  
reistijdenmatrix[36,28] = 87.7  
reistijdenmatrix[36,29] = 87.7  
reistijdenmatrix[36,30] = 71.7  
reistijdenmatrix[36,31] = 139.9  
reistijdenmatrix[36,32] = 99.9  
reistijdenmatrix[36,33] = 60.6  
reistijdenmatrix[36,34] = 37.2  
reistijdenmatrix[36,35] = 40.4  
reistijdenmatrix[36,36] = 16  
reistijdenmatrix[36,37] = 32.1  
reistijdenmatrix[36,38] = 29.7  
reistijdenmatrix[36,39] = 53.4  
reistijdenmatrix[36,40] = 101.7  
reistijdenmatrix[37,1] = 161.8  
reistijdenmatrix[37,2] = 170.8  
reistijdenmatrix[37,3] = 160.2  
reistijdenmatrix[37,4] = 150.2  
reistijdenmatrix[37,5] = 149.2  
reistijdenmatrix[37,6] = 133.4  
reistijdenmatrix[37,7] = 135.8  
reistijdenmatrix[37,8] = 137.9  
reistijdenmatrix[37,9] = 120.7  
reistijdenmatrix[37,10] = 100.7  
reistijdenmatrix[37,11] = 84.3  
reistijdenmatrix[37,12] = 102.9  
reistijdenmatrix[37,13] = 70.7  
reistijdenmatrix[37,14] = 97  
reistijdenmatrix[37,15] = 51.4  
reistijdenmatrix[37,16] = 67.4  
reistijdenmatrix[37,17] = 88.1  
reistijdenmatrix[37,18] = 141.3  
reistijdenmatrix[37,19] = 134.7



reistijdenmatrix[37,20] = 120.4  
reistijdenmatrix[37,21] = 119  
reistijdenmatrix[37,22] = 117.2  
reistijdenmatrix[37,23] = 113.5  
reistijdenmatrix[37,24] = 96  
reistijdenmatrix[37,25] = 121.4  
reistijdenmatrix[37,26] = 116.6  
reistijdenmatrix[37,27] = 118.4  
reistijdenmatrix[37,28] = 106.6  
reistijdenmatrix[37,29] = 111.5  
reistijdenmatrix[37,30] = 89.8  
reistijdenmatrix[37,31] = 165.6  
reistijdenmatrix[37,32] = 125.6  
reistijdenmatrix[37,33] = 86.3  
reistijdenmatrix[37,34] = 62.9  
reistijdenmatrix[37,35] = 49.8  
reistijdenmatrix[37,36] = 32.1  
reistijdenmatrix[37,37] = 12  
reistijdenmatrix[37,38] = 35.9  
reistijdenmatrix[37,39] = 68  
reistijdenmatrix[37,40] = 116.5  
reistijdenmatrix[38,1] = 191.3  
reistijdenmatrix[38,2] = 200.3  
reistijdenmatrix[38,3] = 189.7  
reistijdenmatrix[38,4] = 179.7  
reistijdenmatrix[38,5] = 167.6  
reistijdenmatrix[38,6] = 162.9  
reistijdenmatrix[38,7] = 165.3  
reistijdenmatrix[38,8] = 167.4  
reistijdenmatrix[38,9] = 150.2  
reistijdenmatrix[38,10] = 130.2  
reistijdenmatrix[38,11] = 113.8  
reistijdenmatrix[38,12] = 132.4  
reistijdenmatrix[38,13] = 100.2  
reistijdenmatrix[38,14] = 126.5  
reistijdenmatrix[38,15] = 80.9  
reistijdenmatrix[38,16] = 66.2  
reistijdenmatrix[38,17] = 89.9  
reistijdenmatrix[38,18] = 141.2  
reistijdenmatrix[38,19] = 134.6  
reistijdenmatrix[38,20] = 120.2  
reistijdenmatrix[38,21] = 118.8  
reistijdenmatrix[38,22] = 117.2  
reistijdenmatrix[38,23] = 113.4  
reistijdenmatrix[38,24] = 100.9  
reistijdenmatrix[38,25] = 121.3  
reistijdenmatrix[38,26] = 114.4  
reistijdenmatrix[38,27] = 113  
reistijdenmatrix[38,28] = 106.4  
reistijdenmatrix[38,29] = 106.2  
reistijdenmatrix[38,30] = 90.4  
reistijdenmatrix[38,31] = 158.5  
reistijdenmatrix[38,32] = 118.5  
reistijdenmatrix[38,33] = 79.2  
reistijdenmatrix[38,34] = 55.9

reistijdenmatrix[38,35] = 61.7  
reistijdenmatrix[38,36] = 29.7  
reistijdenmatrix[38,37] = 35.9  
reistijdenmatrix[38,38] = 10  
reistijdenmatrix[38,39] = 35.9  
reistijdenmatrix[38,40] = 120.4  
reistijdenmatrix[39,1] = 220  
reistijdenmatrix[39,2] = 228.9  
reistijdenmatrix[39,3] = 218.4  
reistijdenmatrix[39,4] = 204.2  
reistijdenmatrix[39,5] = 191.3  
reistijdenmatrix[39,6] = 191.6  
reistijdenmatrix[39,7] = 194  
reistijdenmatrix[39,8] = 196.1  
reistijdenmatrix[39,9] = 178.9  
reistijdenmatrix[39,10] = 158.9  
reistijdenmatrix[39,11] = 142.5  
reistijdenmatrix[39,12] = 161.1  
reistijdenmatrix[39,13] = 128.9  
reistijdenmatrix[39,14] = 155.2  
reistijdenmatrix[39,15] = 109.5  
reistijdenmatrix[39,16] = 89.9  
reistijdenmatrix[39,17] = 113.6  
reistijdenmatrix[39,18] = 164.9  
reistijdenmatrix[39,19] = 158.3  
reistijdenmatrix[39,20] = 143.9  
reistijdenmatrix[39,21] = 142.5  
reistijdenmatrix[39,22] = 140.9  
reistijdenmatrix[39,23] = 137.2  
reistijdenmatrix[39,24] = 124.6  
reistijdenmatrix[39,25] = 145  
reistijdenmatrix[39,26] = 138.2  
reistijdenmatrix[39,27] = 136.7  
reistijdenmatrix[39,28] = 130.1  
reistijdenmatrix[39,29] = 129.9  
reistijdenmatrix[39,30] = 114.1  
reistijdenmatrix[39,31] = 182.2  
reistijdenmatrix[39,32] = 142.2  
reistijdenmatrix[39,33] = 102.9  
reistijdenmatrix[39,34] = 79.6  
reistijdenmatrix[39,35] = 85.4  
reistijdenmatrix[39,36] = 53.4  
reistijdenmatrix[39,37] = 68  
reistijdenmatrix[39,38] = 35.9  
reistijdenmatrix[39,39] = 16  
reistijdenmatrix[39,40] = 144.1  
reistijdenmatrix[40,1] = 112.2  
reistijdenmatrix[40,2] = 104.1  
reistijdenmatrix[40,3] = 93.5  
reistijdenmatrix[40,4] = 67.5  
reistijdenmatrix[40,5] = 54.7  
reistijdenmatrix[40,6] = 58.1  
reistijdenmatrix[40,7] = 92.2  
reistijdenmatrix[40,8] = 94.6  
reistijdenmatrix[40,9] = 77.1

```

reistijdenmatrix[40,10] = 57.2
reistijdenmatrix[40,11] = 70.3
reistijdenmatrix[40,12] = 89
reistijdenmatrix[40,13] = 51.1
reistijdenmatrix[40,14] = 94.8
reistijdenmatrix[40,15] = 85
reistijdenmatrix[40,16] = 62.9
reistijdenmatrix[40,17] = 43.8
reistijdenmatrix[40,18] = 61
reistijdenmatrix[40,19] = 67.6
reistijdenmatrix[40,20] = 55.8
reistijdenmatrix[40,21] = 54.6
reistijdenmatrix[40,22] = 45.1
reistijdenmatrix[40,23] = 47.6
reistijdenmatrix[40,24] = 32.1
reistijdenmatrix[40,25] = 56.5
reistijdenmatrix[40,26] = 70.5
reistijdenmatrix[40,27] = 77.9
reistijdenmatrix[40,28] = 69.5
reistijdenmatrix[40,29] = 94.4
reistijdenmatrix[40,30] = 69.2
reistijdenmatrix[40,31] = 174.3
reistijdenmatrix[40,32] = 134.4
reistijdenmatrix[40,33] = 96.7
reistijdenmatrix[40,34] = 89.8
reistijdenmatrix[40,35] = 87.4
reistijdenmatrix[40,36] = 101.7
reistijdenmatrix[40,37] = 116.5
reistijdenmatrix[40,38] = 120.4
reistijdenmatrix[40,39] = 144.1
reistijdenmatrix[40,40] = 21
pct_Pendel[van,naar] = GRAPH(reistijd[van,naar]/reisbereidheid)
(0.00, 0.995), (0.2, 0.995), (0.4, 0.995), (0.6, 0.995), (0.8, 0.995), (1.00, 0.97), (1.20, 0.895),
(1.40, 0.58), (1.60, 0.28), (1.80, 0.155), (2.00, 0.085), (2.20, 0.035), (2.40, 0.01), (2.60, 0.00),
(2.80, 0.00), (3.00, 0.00), (3.20, 0.00), (3.40, 0.00), (3.60, 0.00), (3.80, 0.00), (4.00, 0.00), (4.20,
0.00), (4.40, 0.00), (4.60, 0.00), (4.80, 0.00), (5.00, 0.00), (5.20, 0.00), (5.40, 0.00), (5.60, 0.00),
(5.80, 0.00), (6.00, 0.00)
reisbereidheid = GRAPH(TIME)
(1970, 60.0), (2030, 70.0)
reistijdontwikkeling = GRAPH(TIME)
(1970, 1.04), (1976, 1.04), (1982, 1.02), (1988, 1.01), (1994, 0.98), (2000, 0.97), (2006, 0.96),
(2012, 0.95), (2018, 0.95), (2024, 0.95), (2030, 0.95)
WONINGVOORRAAD[1](t) = WONINGVOORRAAD[1](t - dt) + (Woningbouw[1] -
Woningsloop[1]) * dt
INIT WONINGVOORRAAD[1] = 45115
WONINGVOORRAAD[2](t) = WONINGVOORRAAD[2](t - dt) + (Woningbouw[2] -
Woningsloop[2]) * dt
INIT WONINGVOORRAAD[2] = 15540
WONINGVOORRAAD[3](t) = WONINGVOORRAAD[3](t - dt) + (Woningbouw[3] -
Woningsloop[3]) * dt
INIT WONINGVOORRAAD[3] = 100500
WONINGVOORRAAD[4](t) = WONINGVOORRAAD[4](t - dt) + (Woningbouw[4] -
Woningsloop[4]) * dt
INIT WONINGVOORRAAD[4] = 86000

```

WONINGVOORRAAD[5](t) = WONINGVOORRAAD[5](t - dt) + (Woningbouw[5] -  
Woningsloop[5]) \* dt  
INIT WONINGVOORRAAD[5] = 25125  
WONINGVOORRAAD[6](t) = WONINGVOORRAAD[6](t - dt) + (Woningbouw[6] -  
Woningsloop[6]) \* dt  
INIT WONINGVOORRAAD[6] = 45550  
WONINGVOORRAAD[7](t) = WONINGVOORRAAD[7](t - dt) + (Woningbouw[7] -  
Woningsloop[7]) \* dt  
INIT WONINGVOORRAAD[7] = 37000  
WONINGVOORRAAD[8](t) = WONINGVOORRAAD[8](t - dt) + (Woningbouw[8] -  
Woningsloop[8]) \* dt  
INIT WONINGVOORRAAD[8] = 37475  
WONINGVOORRAAD[9](t) = WONINGVOORRAAD[9](t - dt) + (Woningbouw[9] -  
Woningsloop[9]) \* dt  
INIT WONINGVOORRAAD[9] = 27165  
WONINGVOORRAAD[10](t) = WONINGVOORRAAD[10](t - dt) + (Woningbouw[10] -  
Woningsloop[10]) \* dt  
INIT WONINGVOORRAAD[10] = 71000  
WONINGVOORRAAD[11](t) = WONINGVOORRAAD[11](t - dt) + (Woningbouw[11] -  
Woningsloop[11]) \* dt  
INIT WONINGVOORRAAD[11] = 31800  
WONINGVOORRAAD[12](t) = WONINGVOORRAAD[12](t - dt) + (Woningbouw[12] -  
Woningsloop[12]) \* dt  
INIT WONINGVOORRAAD[12] = 135100  
WONINGVOORRAAD[13](t) = WONINGVOORRAAD[13](t - dt) + (Woningbouw[13] -  
Woningsloop[13]) \* dt  
INIT WONINGVOORRAAD[13] = 123600  
WONINGVOORRAAD[14](t) = WONINGVOORRAAD[14](t - dt) + (Woningbouw[14] -  
Woningsloop[14]) \* dt  
INIT WONINGVOORRAAD[14] = 80000  
WONINGVOORRAAD[15](t) = WONINGVOORRAAD[15](t - dt) + (Woningbouw[15] -  
Woningsloop[15]) \* dt  
INIT WONINGVOORRAAD[15] = 151050  
WONINGVOORRAAD[16](t) = WONINGVOORRAAD[16](t - dt) + (Woningbouw[16] -  
Woningsloop[16]) \* dt  
INIT WONINGVOORRAAD[16] = 43300  
WONINGVOORRAAD[17](t) = WONINGVOORRAAD[17](t - dt) + (Woningbouw[17] -  
Woningsloop[17]) \* dt  
INIT WONINGVOORRAAD[17] = 231050  
WONINGVOORRAAD[18](t) = WONINGVOORRAAD[18](t - dt) + (Woningbouw[18] -  
Woningsloop[18]) \* dt  
INIT WONINGVOORRAAD[18] = 65000  
WONINGVOORRAAD[19](t) = WONINGVOORRAAD[19](t - dt) + (Woningbouw[19] -  
Woningsloop[19]) \* dt  
INIT WONINGVOORRAAD[19] = 40425  
WONINGVOORRAAD[20](t) = WONINGVOORRAAD[20](t - dt) + (Woningbouw[20] -  
Woningsloop[20]) \* dt  
INIT WONINGVOORRAAD[20] = 44250  
WONINGVOORRAAD[21](t) = WONINGVOORRAAD[21](t - dt) + (Woningbouw[21] -  
Woningsloop[21]) \* dt  
INIT WONINGVOORRAAD[21] = 72625  
WONINGVOORRAAD[22](t) = WONINGVOORRAAD[22](t - dt) + (Woningbouw[22] -  
Woningsloop[22]) \* dt  
INIT WONINGVOORRAAD[22] = 40000

$WONINGVOORRAAD[23](t) = WONINGVOORRAAD[23](t - dt) + (Woningbouw[23] - Woningloop[23]) * dt$   
 INIT  $WONINGVOORRAAD[23] = 361700$   
 $WONINGVOORRAAD[24](t) = WONINGVOORRAAD[24](t - dt) + (Woningbouw[24] - Woningloop[24]) * dt$   
 INIT  $WONINGVOORRAAD[24] = 68525$   
 $WONINGVOORRAAD[25](t) = WONINGVOORRAAD[25](t - dt) + (Woningbouw[25] - Woningloop[25]) * dt$   
 INIT  $WONINGVOORRAAD[25] = 82000$   
 $WONINGVOORRAAD[26](t) = WONINGVOORRAAD[26](t - dt) + (Woningbouw[26] - Woningloop[26]) * dt$   
 INIT  $WONINGVOORRAAD[26] = 230875$   
 $WONINGVOORRAAD[27](t) = WONINGVOORRAAD[27](t - dt) + (Woningbouw[27] - Woningloop[27]) * dt$   
 INIT  $WONINGVOORRAAD[27] = 51200$   
 $WONINGVOORRAAD[28](t) = WONINGVOORRAAD[28](t - dt) + (Woningbouw[28] - Woningloop[28]) * dt$   
 INIT  $WONINGVOORRAAD[28] = 60000$   
 $WONINGVOORRAAD[29](t) = WONINGVOORRAAD[29](t - dt) + (Woningbouw[29] - Woningloop[29]) * dt$   
 INIT  $WONINGVOORRAAD[29] = 395000$   
 $WONINGVOORRAAD[30](t) = WONINGVOORRAAD[30](t - dt) + (Woningbouw[30] - Woningloop[30]) * dt$   
 INIT  $WONINGVOORRAAD[30] = 95000$   
 $WONINGVOORRAAD[31](t) = WONINGVOORRAAD[31](t - dt) + (Woningbouw[31] - Woningloop[31]) * dt$   
 INIT  $WONINGVOORRAAD[31] = 32300$   
 $WONINGVOORRAAD[32](t) = WONINGVOORRAAD[32](t - dt) + (Woningbouw[32] - Woningloop[32]) * dt$   
 INIT  $WONINGVOORRAAD[32] = 65700$   
 $WONINGVOORRAAD[33](t) = WONINGVOORRAAD[33](t - dt) + (Woningbouw[33] - Woningloop[33]) * dt$   
 INIT  $WONINGVOORRAAD[33] = 121800$   
 $WONINGVOORRAAD[34](t) = WONINGVOORRAAD[34](t - dt) + (Woningbouw[34] - Woningloop[34]) * dt$   
 INIT  $WONINGVOORRAAD[34] = 88000$   
 $WONINGVOORRAAD[35](t) = WONINGVOORRAAD[35](t - dt) + (Woningbouw[35] - Woningloop[35]) * dt$   
 INIT  $WONINGVOORRAAD[35] = 105000$   
 $WONINGVOORRAAD[36](t) = WONINGVOORRAAD[36](t - dt) + (Woningbouw[36] - Woningloop[36]) * dt$   
 INIT  $WONINGVOORRAAD[36] = 140000$   
 $WONINGVOORRAAD[37](t) = WONINGVOORRAAD[37](t - dt) + (Woningbouw[37] - Woningloop[37]) * dt$   
 INIT  $WONINGVOORRAAD[37] = 52000$   
 $WONINGVOORRAAD[38](t) = WONINGVOORRAAD[38](t - dt) + (Woningbouw[38] - Woningloop[38]) * dt$   
 INIT  $WONINGVOORRAAD[38] = 44305$   
 $WONINGVOORRAAD[39](t) = WONINGVOORRAAD[39](t - dt) + (Woningbouw[39] - Woningloop[39]) * dt$   
 INIT  $WONINGVOORRAAD[39] = 162200$   
 $WONINGVOORRAAD[40](t) = WONINGVOORRAAD[40](t - dt) + (Woningbouw[40] - Woningloop[40]) * dt$   
 INIT  $WONINGVOORRAAD[40] = 14000$   
 $Woningbouw[van] = WONINGVOORRAAD[van]*woningbouwfactor[van]$

Woningsloop[van] = (WONINGVOORRAAD[van])\*woningsloopfactor[van]  
 NR\_woningbouw[1] = 0.017  
 NR\_woningbouw[2] = 0.016  
 NR\_woningbouw[3] = 0.017  
 NR\_woningbouw[4] = 0.018  
 NR\_woningbouw[5] = 0.021  
 NR\_woningbouw[6] = 0.022  
 NR\_woningbouw[7] = 0.025  
 NR\_woningbouw[8] = 0.023  
 NR\_woningbouw[9] = 0.023  
 NR\_woningbouw[10] = 0.023  
 NR\_woningbouw[11] = 0.020  
 NR\_woningbouw[12] = 0.022  
 NR\_woningbouw[13] = 0.023  
 NR\_woningbouw[14] = 0.022  
 NR\_woningbouw[15] = 0.023  
 NR\_woningbouw[16] = 0.023  
 NR\_woningbouw[17] = 0.024  
 NR\_woningbouw[18] = 0.028  
 NR\_woningbouw[19] = 0.029  
 NR\_woningbouw[20] = 0.018  
 NR\_woningbouw[21] = 0.010  
 NR\_woningbouw[22] = 0.018  
 NR\_woningbouw[23] = 0.017  
 NR\_woningbouw[24] = 0.014  
 NR\_woningbouw[25] = 0.022  
 NR\_woningbouw[26] = 0.015  
 NR\_woningbouw[27] = 0.022  
 NR\_woningbouw[28] = 0.027  
 NR\_woningbouw[29] = 0.018  
 NR\_woningbouw[30] = 0.021  
 NR\_woningbouw[31] = 0.017  
 NR\_woningbouw[32] = 0.020  
 NR\_woningbouw[33] = 0.025  
 NR\_woningbouw[34] = 0.024  
 NR\_woningbouw[35] = 0.028  
 NR\_woningbouw[36] = 0.026  
 NR\_woningbouw[37] = 0.026  
 NR\_woningbouw[38] = 0.024  
 NR\_woningbouw[39] = 0.019  
 NR\_woningbouw[40] = 0.078  
 correctiefactor\_woningbouw[van] = GRAPH(TIME)  
 (1970, 1.42), (1971, 1.64), (1972, 1.79), (1973, 1.76), (1974, 1.62), (1975, 1.34), (1976, 1.19),  
 (1977, 1.25), (1978, 1.14), (1979, 0.885), (1980, 1.11), (1981, 1.14), (1982, 1.11), (1983, 0.905),  
 (1984, 0.903), (1985, 0.82), (1986, 0.855), (1987, 0.874), (1988, 0.919), (1989, 0.874), (1990,  
 0.748), (1991, 0.618), (1992, 0.626), (1993, 0.616), (1994, 0.647), (1995, 0.689), (1996, 0.633),  
 (1997, 0.663), (1998, 0.621), (1999, 0.538), (2000, 0.621), (2001, 0.582), (2002, 0.541), (2003,  
 0.518), (2004, 0.486), (2005, 0.365), (2006, 0.36), (2007, 0.353), (2008, 0.349), (2009, 0.349),  
 (2010, 0.352), (2011, 0.357), (2012, 0.361), (2013, 0.363), (2014, 0.366), (2015, 0.375), (2016,  
 0.377), (2017, 0.372), (2018, 0.361), (2019, 0.354), (2020, 0.352), (2021, 0.351), (2022, 0.349),  
 (2023, 0.347), (2024, 0.345), (2025, 0.344), (2026, 0.342), (2027, 0.34), (2028, 0.338), (2029,  
 0.337), (2030, 0.34)  
 HHWR[van] = Huishoudens[van]/WONINGVOORRAAD[van]  
 huishoudensgrootte[1] = 3.2  
 huishoudensgrootte[2] = 3.3

huishoudensgrootte[3] = 3.5  
huishoudensgrootte[4] = 3.25  
huishoudensgrootte[5] = 3.39  
huishoudensgrootte[6] = 3.28  
huishoudensgrootte[7] = 3.45  
huishoudensgrootte[8] = 3.65  
huishoudensgrootte[9] = 3.5  
huishoudensgrootte[10] = 3.67  
huishoudensgrootte[11] = 3.59  
huishoudensgrootte[12] = 3.71  
huishoudensgrootte[13] = 3.79  
huishoudensgrootte[14] = 3.89  
huishoudensgrootte[15] = 3.7  
huishoudensgrootte[16] = 3.55  
huishoudensgrootte[17] = 3.53  
huishoudensgrootte[18] = 3.56  
huishoudensgrootte[19] = 3.5  
huishoudensgrootte[20] = 3.59  
huishoudensgrootte[21] = 3.28  
huishoudensgrootte[22] = 3.25  
huishoudensgrootte[23] = 2.95  
huishoudensgrootte[24] = 3.38  
huishoudensgrootte[25] = 3.69  
huishoudensgrootte[26] = 3.01  
huishoudensgrootte[27] = 3.54  
huishoudensgrootte[28] = 3.53  
huishoudensgrootte[29] = 3.01  
huishoudensgrootte[30] = 3.3  
huishoudensgrootte[31] = 3.05  
huishoudensgrootte[32] = 3.03  
huishoudensgrootte[33] = 3.61  
huishoudensgrootte[34] = 3.75  
huishoudensgrootte[35] = 4.2  
huishoudensgrootte[36] = 3.88  
huishoudensgrootte[37] = 4.07  
huishoudensgrootte[38] = 3.98  
huishoudensgrootte[39] = 3.6  
huishoudensgrootte[40] = 4.4  
NR\_woningsloop[1] = 0.00465  
NR\_woningsloop[2] = 0.00386  
NR\_woningsloop[3] = 0.00319  
NR\_woningsloop[4] = 0.00363  
NR\_woningsloop[5] = 0.00393  
NR\_woningsloop[6] = 0.00256  
NR\_woningsloop[7] = 0.00215  
NR\_woningsloop[8] = 0.00247  
NR\_woningsloop[9] = 0.00199  
NR\_woningsloop[10] = 0.00199  
NR\_woningsloop[11] = 0.00212  
NR\_woningsloop[12] = 0.00283  
NR\_woningsloop[13] = 0.00223  
NR\_woningsloop[14] = 0.00194  
NR\_woningsloop[15] = 0.00200  
NR\_woningsloop[16] = 0.00169  
NR\_woningsloop[17] = 0.00174

NR\_woningsloop[18] = 0.00163  
NR\_woningsloop[19] = 0.00139  
NR\_woningsloop[20] = 0.00134  
NR\_woningsloop[21] = 0.00186  
NR\_woningsloop[22] = 0.00271  
NR\_woningsloop[23] = 0.00403  
NR\_woningsloop[24] = 0.00104  
NR\_woningsloop[25] = 0.00197  
NR\_woningsloop[26] = 0.00351  
NR\_woningsloop[27] = 0.00257  
NR\_woningsloop[28] = 0.00245  
NR\_woningsloop[29] = 0.00476  
NR\_woningsloop[30] = 0.00301  
NR\_woningsloop[31] = 0.00308  
NR\_woningsloop[32] = 0.00334  
NR\_woningsloop[33] = 0.00177  
NR\_woningsloop[34] = 0.00192  
NR\_woningsloop[35] = 0.00210  
NR\_woningsloop[36] = 0.00201  
NR\_woningsloop[37] = 0.00222  
NR\_woningsloop[38] = 0.00184  
NR\_woningsloop[39] = 0.00204  
NR\_woningsloop[40] = 0.0006  
pct\_EM\_BV[1] = 0.022  
pct\_EM\_BV[2] = 0.02  
pct\_EM\_BV[3] = 0.01  
pct\_EM\_BV[4] = 0.01  
pct\_EM\_BV[5] = 0.01  
pct\_EM\_BV[6] = 0.01  
pct\_EM\_BV[7] = 0.01  
pct\_EM\_BV[8] = 0.02  
pct\_EM\_BV[9] = 0.01  
pct\_EM\_BV[10] = 0.02  
pct\_EM\_BV[11] = 0.01  
pct\_EM\_BV[12] = 0.02  
pct\_EM\_BV[13] = 0.01  
pct\_EM\_BV[14] = 0.02  
pct\_EM\_BV[15] = 0.03  
pct\_EM\_BV[16] = 0.01  
pct\_EM\_BV[17] = 0.01  
pct\_EM\_BV[18] = 0.01  
pct\_EM\_BV[19] = 0.01  
pct\_EM\_BV[20] = 0.015  
pct\_EM\_BV[21] = 0.01  
pct\_EM\_BV[22] = 0.03  
pct\_EM\_BV[23] = 0.01  
pct\_EM\_BV[24] = 0.01  
pct\_EM\_BV[25] = 0.01  
pct\_EM\_BV[26] = 0.01  
pct\_EM\_BV[27] = 0.01  
pct\_EM\_BV[28] = 0.01  
pct\_EM\_BV[29] = 0.015  
pct\_EM\_BV[30] = 0.01  
pct\_EM\_BV[31] = 0.025  
pct\_EM\_BV[32] = 0.01



pct\_EM\_BV[33] = 0.02  
pct\_EM\_BV[34] = 0.02  
pct\_EM\_BV[35] = 0.01  
pct\_EM\_BV[36] = 0.02  
pct\_EM\_BV[37] = 0.02  
pct\_EM\_BV[38] = 0.02  
pct\_EM\_BV[39] = 0.03  
pct\_EM\_BV[40] = 0.005  
pct\_IMM\_BV[1] = 0.02  
pct\_IMM\_BV[2] = 0.02  
pct\_IMM\_BV[3] = 0.01  
pct\_IMM\_BV[4] = 0.01  
pct\_IMM\_BV[5] = 0.01  
pct\_IMM\_BV[6] = 0.03  
pct\_IMM\_BV[7] = 0.02  
pct\_IMM\_BV[8] = 0.02  
pct\_IMM\_BV[9] = 0.01  
pct\_IMM\_BV[10] = 0.02  
pct\_IMM\_BV[11] = 0.01  
pct\_IMM\_BV[12] = 0.02  
pct\_IMM\_BV[13] = 0.01  
pct\_IMM\_BV[14] = 0.02  
pct\_IMM\_BV[15] = 0.02  
pct\_IMM\_BV[16] = 0.01  
pct\_IMM\_BV[17] = 0.01  
pct\_IMM\_BV[18] = 0.01  
pct\_IMM\_BV[19] = 0.01  
pct\_IMM\_BV[20] = 0.01  
pct\_IMM\_BV[21] = 0.005  
pct\_IMM\_BV[22] = 0.005  
pct\_IMM\_BV[23] = 0.01  
pct\_IMM\_BV[24] = 0.005  
pct\_IMM\_BV[25] = 0.005  
pct\_IMM\_BV[26] = 0.005  
pct\_IMM\_BV[27] = 0.01  
pct\_IMM\_BV[28] = 0.01  
pct\_IMM\_BV[29] = 0.01  
pct\_IMM\_BV[30] = 0.01  
pct\_IMM\_BV[31] = 0.03  
pct\_IMM\_BV[32] = 0.01  
pct\_IMM\_BV[33] = 0.025  
pct\_IMM\_BV[34] = 0.02  
pct\_IMM\_BV[35] = 0.01  
pct\_IMM\_BV[36] = 0.02  
pct\_IMM\_BV[37] = 0.02  
pct\_IMM\_BV[38] = 0.02  
pct\_IMM\_BV[39] = 0.025  
pct\_IMM\_BV[40] = 0.02  
pct\_pot\_bb[van] = TIME  
pct\_RESINM[1] = 0.07  
pct\_RESINM[2] = 0.05  
pct\_RESINM[3] = 0.05  
pct\_RESINM[4] = 0.08  
pct\_RESINM[5] = 0.07  
pct\_RESINM[6] = 0.06

pct\_RESINM[7] = 0.05  
pct\_RESINM[8] = 0.03  
pct\_RESINM[9] = 0.03  
pct\_RESINM[10] = 0.03  
pct\_RESINM[11] = 0.03  
pct\_RESINM[12] = 0.03  
pct\_RESINM[13] = 0.03  
pct\_RESINM[14] = 0.03  
pct\_RESINM[15] = 0.03  
pct\_RESINM[16] = 0.03  
pct\_RESINM[17] = 0.0297  
pct\_RESINM[18] = 0.03  
pct\_RESINM[19] = 0.03  
pct\_RESINM[20] = 0.01  
pct\_RESINM[21] = 0.01  
pct\_RESINM[22] = 0.01  
pct\_RESINM[23] = 0.01  
pct\_RESINM[24] = 0.02  
pct\_RESINM[25] = 0.02  
pct\_RESINM[26] = 0.01  
pct\_RESINM[27] = 0.01  
pct\_RESINM[28] = 0.03  
pct\_RESINM[29] = 0.01  
pct\_RESINM[30] = 0.03  
pct\_RESINM[31] = 0.03  
pct\_RESINM[32] = 0.03  
pct\_RESINM[33] = 0.03  
pct\_RESINM[34] = 0.03  
pct\_RESINM[35] = 0.03  
pct\_RESINM[36] = 0.03  
pct\_RESINM[37] = 0.03  
pct\_RESINM[38] = 0.03  
pct\_RESINM[39] = 0.03  
pct\_RESINM[40] = 0.09  
pct\_RESUITM[1] = 0.03  
pct\_RESUITM[2] = 0.05  
pct\_RESUITM[3] = 0.06  
pct\_RESUITM[4] = 0.03  
pct\_RESUITM[5] = 0.03  
pct\_RESUITM[6] = 0.03  
pct\_RESUITM[7] = 0.03  
pct\_RESUITM[8] = 0.03  
pct\_RESUITM[9] = 0.03  
pct\_RESUITM[10] = 0.03  
pct\_RESUITM[11] = 0.03  
pct\_RESUITM[12] = 0.03  
pct\_RESUITM[13] = 0.03  
pct\_RESUITM[14] = 0.01  
pct\_RESUITM[15] = 0.03  
pct\_RESUITM[16] = 0.03  
pct\_RESUITM[17] = 0.065  
pct\_RESUITM[18] = 0.03  
pct\_RESUITM[19] = 0.03  
pct\_RESUITM[20] = 0.03  
pct\_RESUITM[21] = 0.03

pct\_RESUITM[22] = 0.03  
 pct\_RESUITM[23] = 0.03  
 pct\_RESUITM[24] = 0.03  
 pct\_RESUITM[25] = 0.03  
 pct\_RESUITM[26] = 0.03  
 pct\_RESUITM[27] = 0.03  
 pct\_RESUITM[28] = 0.03  
 pct\_RESUITM[29] = 0.03  
 pct\_RESUITM[30] = 0.03  
 pct\_RESUITM[31] = 0.03  
 pct\_RESUITM[32] = 0.03  
 pct\_RESUITM[33] = 0.03  
 pct\_RESUITM[34] = 0.03  
 pct\_RESUITM[35] = 0.03  
 pct\_RESUITM[36] = 0.03  
 pct\_RESUITM[37] = 0.03  
 pct\_RESUITM[38] = 0.03  
 pct\_RESUITM[39] = 0.03  
 pct\_RESUITM[40] = 0.03  
 woningbouwfactor[van] =  
 NR\_woningbouw[van]\*WONHHBM[van]\*WONLANDBM[van]\*correctiefactor\_woningbouw[van]  
 woningsloopfactor[van] =  
 NR\_woningsloop[van]\*correctiefactor\_woningsloop[van]\*WONLANDSM[van]  
 AWONM\_EM\_BEV[van] = GRAPH(HHWR[van])  
 (0.00, 0.36), (0.2, 0.37), (0.4, 0.4), (0.6, 0.48), (0.8, 0.63), (1.00, 1.00), (1.20, 1.38), (1.40, 1.54),  
 (1.60, 1.63), (1.80, 1.64), (2.00, 1.65)  
 AWONM\_IMM\_BEV[van] = GRAPH(HHWR[van])  
 (0.00, 1.60), (0.2, 1.59), (0.4, 1.57), (0.6, 1.47), (0.8, 1.29), (1.00, 1.00), (1.20, 0.72), (1.40, 0.53),  
 (1.60, 0.43), (1.80, 0.41), (2.00, 0.4)  
 BRUIMLANDBM[van] = GRAPH(BL[van])  
 (0.00, 1.00), (0.1, 1.00), (0.2, 1.00), (0.3, 0.98), (0.4, 0.94), (0.5, 0.88), (0.6, 0.81), (0.7, 0.69),  
 (0.8, 0.56), (0.9, 0.36), (1, 0.00)  
 BRUIMLANDSM[van] = GRAPH(BL[van])  
 (0.00, 1.00), (0.1, 1.00), (0.2, 1.00), (0.3, 1.00), (0.4, 1.00), (0.5, 1.00), (0.6, 1.02), (0.7, 1.04),  
 (0.8, 1.08), (0.9, 1.15), (1, 1.29)  
 correctiefactor\_woningsloop[van] = GRAPH(TIME)  
 (1970, 0.00), (1974, 0.00), (1978, 0.00), (1982, 0.00), (1986, 0.00), (1990, 0.00), (1994, 0.00),  
 (1998, 0.00), (2002, 0.00), (2006, 0.00), (2010, 0.00)  
 pct\_pot\_bb[van] = TIME  
 WONHHBM[van] = GRAPH(HHWR[van])  
 (0.00, 0.69), (0.2, 0.7), (0.4, 0.74), (0.6, 0.8), (0.8, 0.89), (1.00, 1.00), (1.20, 1.09), (1.40, 1.20),  
 (1.60, 1.27), (1.80, 1.32), (2.00, 1.33)  
 WONLANDBM[van] = GRAPH(BL[van])  
 (0.00, 1.00), (0.1, 1.00), (0.2, 1.00), (0.3, 0.98), (0.4, 0.95), (0.5, 0.91), (0.6, 0.83), (0.7, 0.69),  
 (0.8, 0.51), (0.9, 0.24), (1, 0.00)  
 WONLANDSM[van] = GRAPH(BL[van])  
 (0.00, 1.00), (0.1, 1.00), (0.2, 1.00), (0.3, 1.00), (0.4, 1.00), (0.5, 1.00), (0.6, 1.02), (0.7, 1.04),  
 (0.8, 1.08), (0.9, 1.15), (1, 1.29)



## **BIJLAGEN HOOFDSTUK 7**

### **MODULE BEVOLKING**

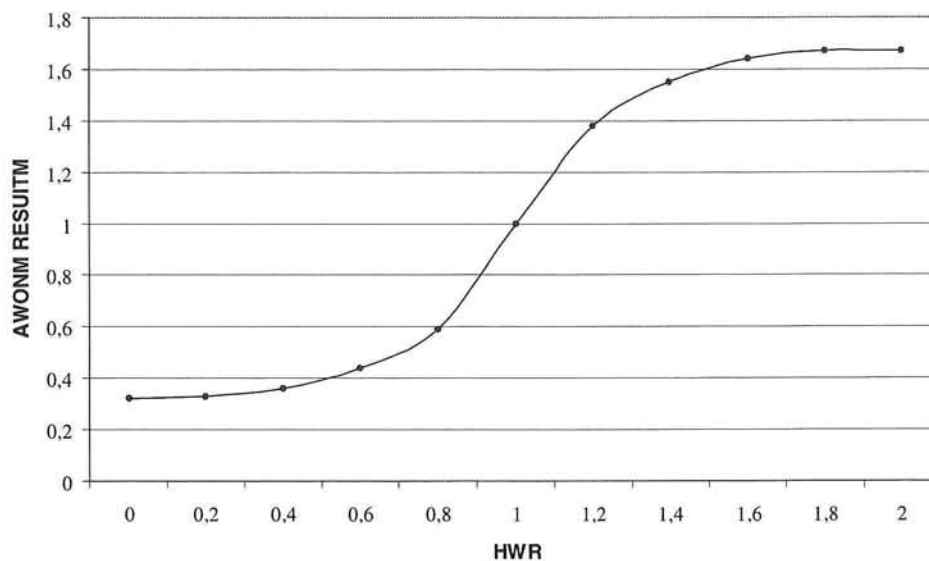


## Bijlage 7.1 Multipliertafels module bevolking

Deze bijlage geeft een overzicht van de gebruikte multipliertafels ten aanzien van het subsysteem bevolking. De gepresenteerde multipliertafels zijn de feitelijke regulatoren in het model, welke zorgen voor de dynamiek tussen subsystemen. Hiermee sturen de multipliertafels de intraregionale en interregionale processen in het model aan. Voor de theoretische achtergrond van de multipliertafels wordt verwezen naar het bijbehorende hoofdstuk in het eindrapport.

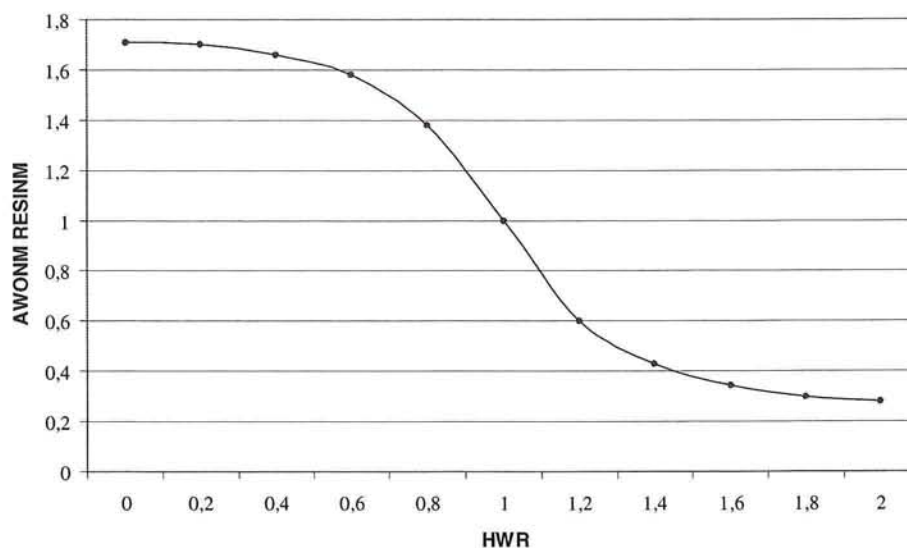
### Bijlage 7.1.1 Multipliertafels residentiële migratie

De aansturing van de potentiële residentiële uitmigratie geschiedt via de multipliertafel *AWONM\_RESUITM* (*attractiviteit van woningen voor potentiële residentiële uitmigratie*). Het verloop van deze multipliertafel is weergegeven in figuur-Bijlage 7.1.



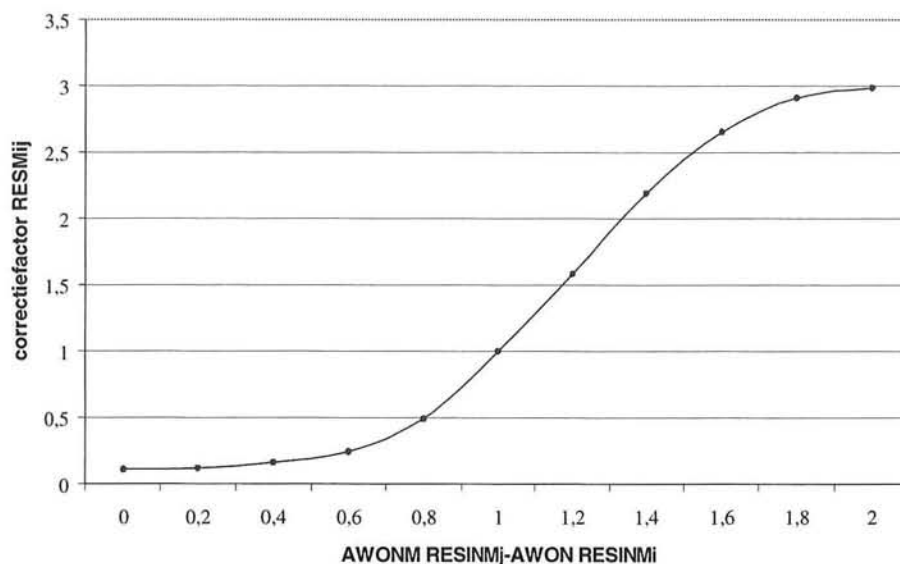
Figuur-Bijlage 7.1 Multipliertafel *AWONM\_RESUITM*

De aansturing van de potentiële residentiële immigratie geschiedt via de multipliertafel *AWON\_RESINM* (*attractiviteit van woningen voor potentiële residentiële immigratie*). Het verloop van deze multipliertafel is weergegeven in figuur-Bijlage 7.2.



Figuur-Bijlage 7.2 Multipliertafel *AWON\_RESINM*

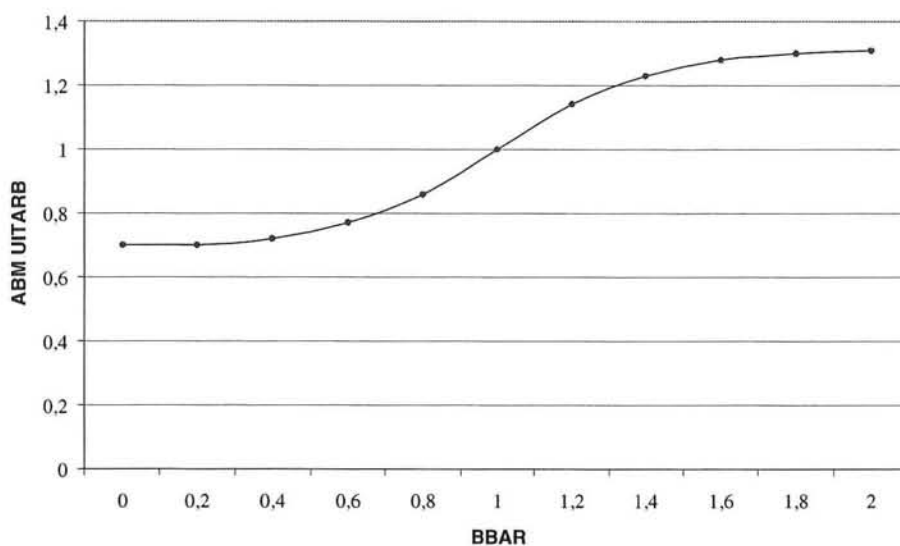
De correctiefactor, welke het effect van verschil in woningmarktattractiviteit op de residentiële migratie weergeeft, is weergegeven in figuur-Bijlage 7.3.



Figuur-Bijlage 7.3 Correctiefactor residentiële migratie

### Bijlage 7.1.2 Multipliertafels arbeidsmigratie en pendel

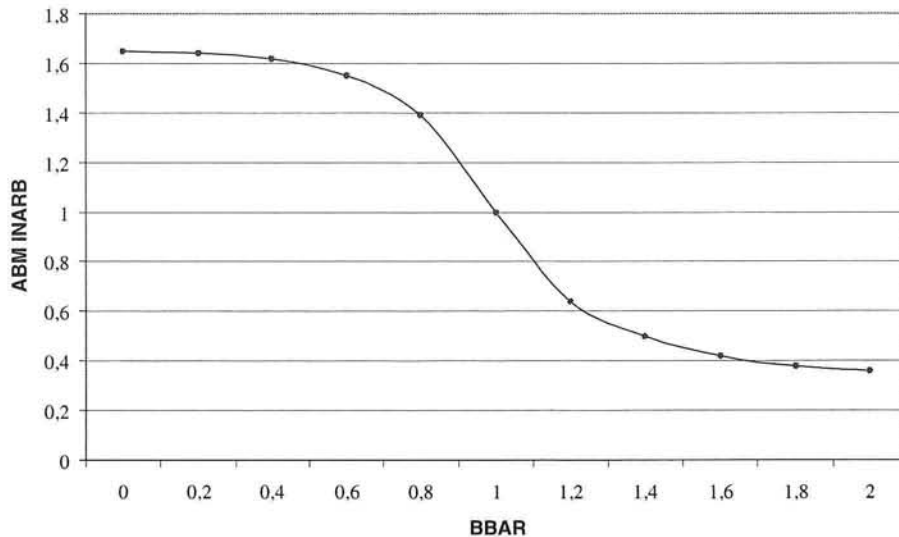
De aansturing van de potentiële arbeidsuitmigratie en uitgaande pendel geschiedt via de multipliertafel *ABM\_UITARB* (*attractiviteit van banen voor potentiële arbeidsuitmigratie/uitgaande pendel*). Het verloop van deze multipliertafel is weergegeven in figuur-Bijlage 7.4.



Figuur-Bijlage 7.4 Multipliertafel ABM\_UITARB

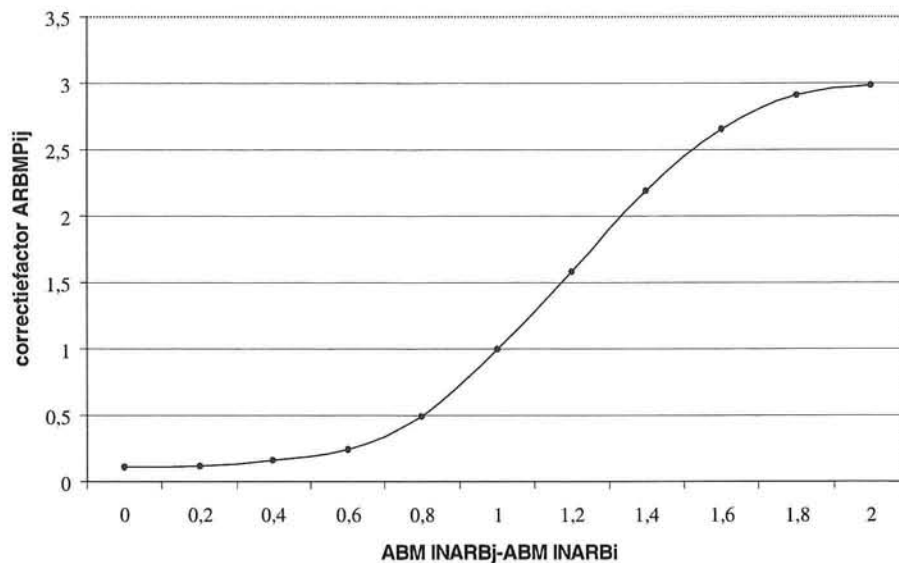
De aansturing van de potentiële arbeidsinmigratie en inkomende pendel geschiedt via de multipliertafel *ABM\_INARBM* (*attractiviteit van banen voor potentiële arbeidsinmigratie en inkomende pendel*). Het verloop van deze multipliertafel is weergegeven in figuur-Bijlage 7.5.





Figuur-Bijlage 7.5 Multipliertafel ABM\_INARB

De correctiefactor, welke het effect van verschil in arbeidsmarktattractiviteit op de arbeidsmigratie en pendel weergeeft, is weergegeven in figuur-Bijlage 7.6.

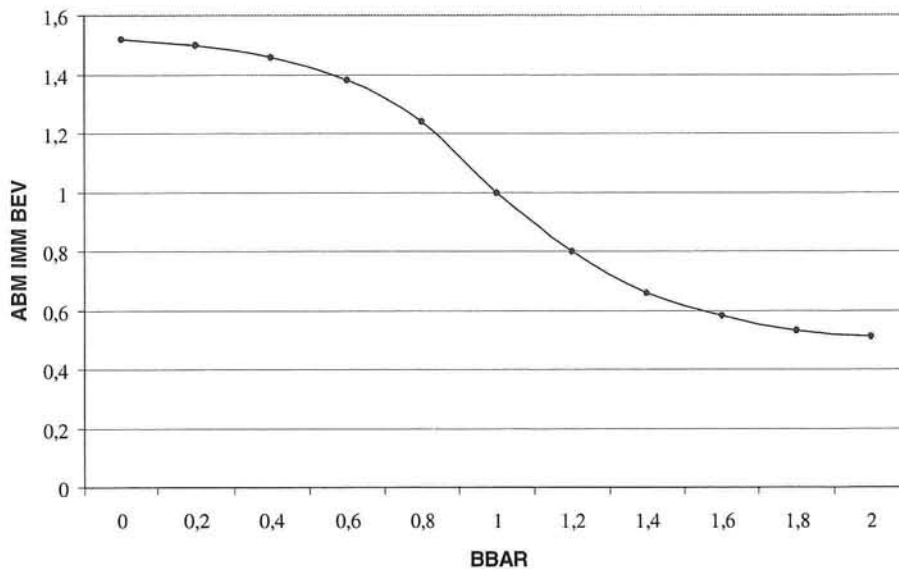


Figuur-Bijlage 7.6 Correctiefactor arbeidsmigratie en pendel

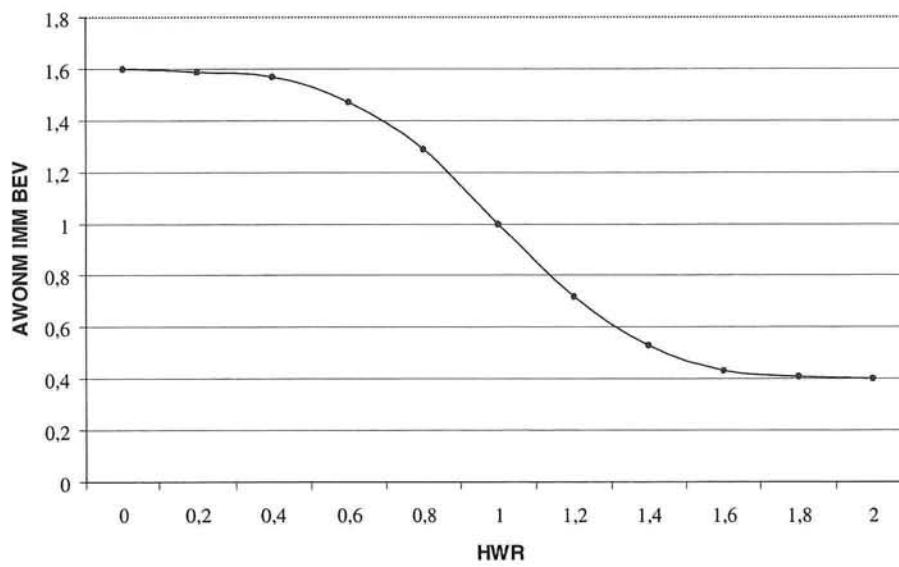
### Bijlage 7.1.3 Multipliertafels buitenlandse migratie bevolking

De aansturing van de buitenlandse immigratie geschiedt via de multipliertafels *AWONM\_IMM\_BEV* (attractiviteit van woningen voor immigranten) en *ABM\_IMM\_BEV* (attractiviteit van banen voor immigranten). Deze multipliertafels zijn weergegeven in figuur-Bijlage 7.7 en figuur-Bijlage 7.8.

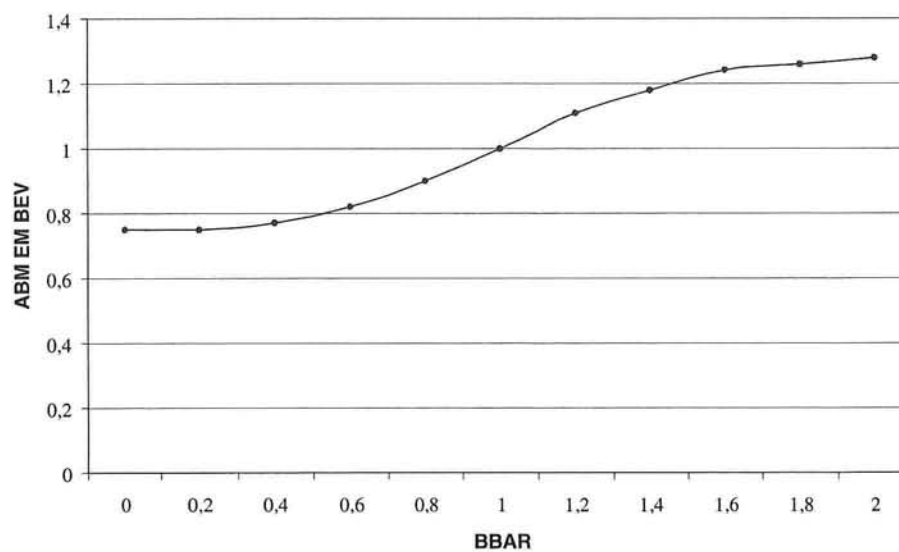
De aansturing van de buitenlandse emigratie geschiedt via de multipliertafels *AWONM\_EM\_BEV* (attractiviteit van woningen voor emigranten) en *ABM\_EM\_BEV* (attractiviteit van banen voor emigranten). Deze multipliertafels zijn weergegeven in figuur-Bijlage 7.9 en figuur-Bijlage 7.10.



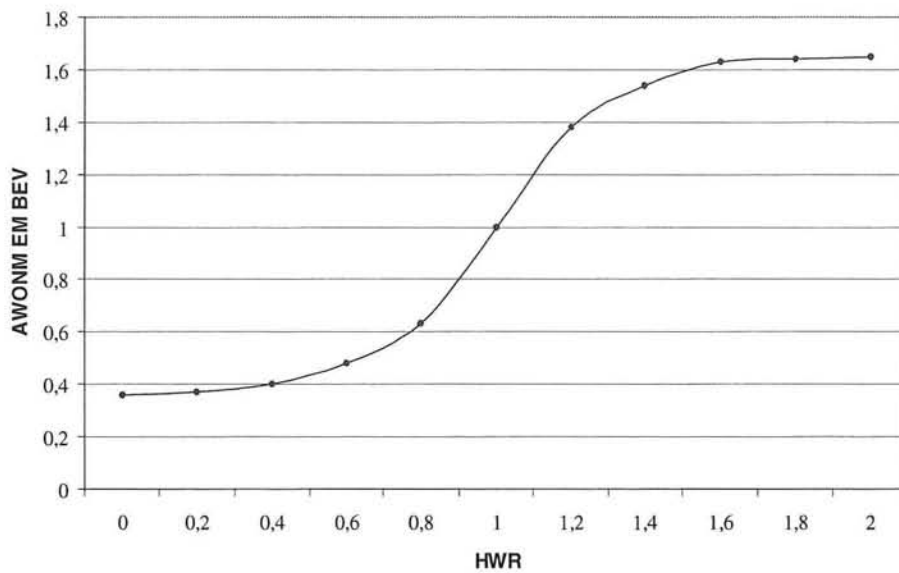
*Figuur-Bijlage 7.7 Multipliertafel ABM\_IMM\_BEV*



*Figuur-Bijlage 7.8 Multipliertafel AWONM\_IMM\_BEV*



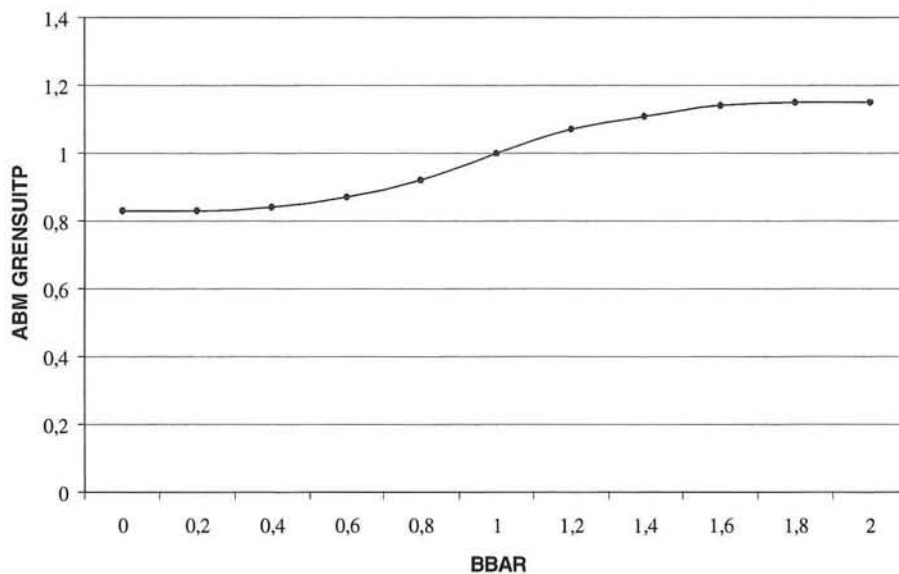
*Figuur-Bijlage 7.9 Multipliertafel ABM\_EM\_BEV*



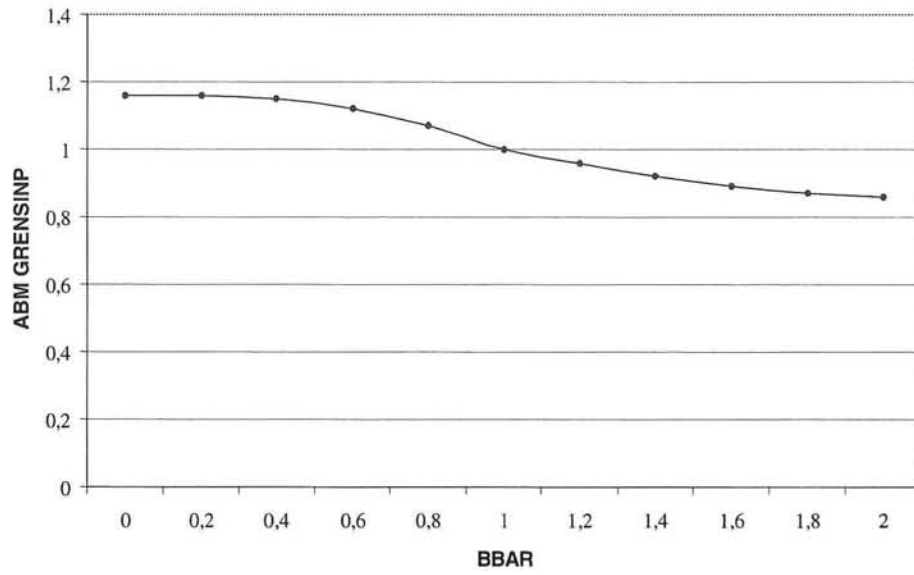
Figuur-Bijlage 7.10 Multipliertafel AWONM\_EM\_BEV

#### Bijlage 7.1.4 Multipliertafels grenspendel

De aansturing van de buitenlandse pendel geschiedt via de multipliertafels *ABM\_GRENSUITP* (*attractiviteit van banen voor uitgaande grenspendel*) en *ABM\_GRENSINP* (*attractiviteit van banen voor inkomende grenspendel*). Deze multipliertafels zijn weergegeven in figuur-Bijlage 7.11 en figuur-Bijlage 7.12.



Figuur-Bijlage 7.11 Multipliertafel ABM\_GRENSUITP



Figuur-Bijlage 7.12 Multipliertafel ABM\_GRENSINP

## Bijlage 7.2 Capaciteitsbeperking migratie

Ten aanzien van migratie kan onderscheid gemaakt worden tussen *willen* (*potentiële* migranten), *kunnen* (dadwerkelijk beschikbare woningen of banen) en *doen* (*daadwerkelijke* migranten). In de huidige modellering is per regio het aantal potentiële migranten in beeld gebracht, die op basis van regionale omstandigheden een woning of baan in een andere regio willen. Aangenomen is dat de mensen die *kunnen* migreren dit ook *doen*. Dat mensen die *willen* ook *kunnen* is maar de vraag. Deze laatste veronderstelling kan aangepast worden. Dit idee wordt hieronder *verkend*, en houdt een verfijning van de migratiemodules in.

Het aantal migranten dat daadwerkelijk *kan* immigreren is onder andere beperkt door het aantal *beschikbare woningen* in de regio van bestemming *j*. In een bestemmingsregio *j* willen vanuit allerlei richtingen mensen immigreren. Er kan in dit verband gesproken worden van **concurrerende migranten**. Het aantal immigranten dat überhaupt in een regio *kan* immigreren kan worden beschreven door:

$$INM_j = \text{if} \left( \text{potINM}_j > \text{beschikbare\_woningen} \right) \text{then} \left( \text{beschikbare\_woningen} \right) \text{else} \left( \text{potINM}_j \right)$$

Waarin:

$INM_j$	= werkelijk totaal aantal immigranten van regio <i>j</i> ;
$\text{potINM}_j$	= aantal potentiële immigranten van regio <i>j</i> o.b.v. relatieve omstandigheden;
$\text{beschikbare\_woningen}_j$	= daadwerkelijk aantal beschikbare woningen in regio <i>j</i> voor immigranten.

Indirect volgt hieruit hoeveel mensen uit regio *i* eventueel toch niet (kunnen en doen) verhuizen naar *j*. Vervolgens kan het aantal woningen, waar door immigranten vraag naar is, worden toegedeeld aan de concurrerende migranten. Naar rato van het aandeel van de vraag van elke regio *i* in de totale vraag in regio *j* worden de beschikbare woningen/immigrant 'uitgedeeld':

$$UITM_{ij} = INM_j * \frac{\text{potUITM}_{ij}}{\text{potINM}_j}$$

Waarin:

$UITM_{ij}$	= werkelijk aantal uitmigranten van regio <i>i</i> naar regio <i>j</i> ;
$INM_j$	= werkelijk totaal aantal immigranten van regio <i>j</i> ;
$\text{potUITM}_{ij}$	= aantal potentiële uitmigranten van regio <i>i</i> naar regio <i>j</i> o.b.v. relatieve omstandigheden;
$\text{potINM}_j$	= aantal potentiële immigranten van regio <i>j</i> o.b.v. relatieve omstandigheden.

Voor regio *i* wordt de mogelijkheid tot interregionaal verhuizen aangeboden/ gaan weg:

Waarin:

$$UITM_i = \sum_j UITM_{ij}$$

$UITM_i$	= werkelijk totaal aantal uitmigranten van regio <i>i</i> ;
$UITM_{ij}$	= werkelijk aantal uitmigranten van regio <i>i</i> naar regio <i>j</i> .

Verondersteld wordt dat het aantal mensen uit regio *i* die de mogelijkheid wordt geboden uiteindelijk migreert. De rest blijft en vergeet de interregionale migratiewens de komende tijdseenheid. Op basis van deze blijvers zou (mede) de woningbouw gestimuleerd kunnen worden. Met enige vertraging (woningbouw moet gereed zijn) kunnen deze mensen dan alsnog migreren. Ook kunnen deze migranten worden verdeeld met een aangepast zwaartekrachtmodel.

## Bijlage 7.3 Desaggregatie van migratie

Door desaggregatie van de bevolking kan de bevolking demografisch beter worden beschreven. Desaggregatie van de bevolking geeft ook ten aanzien van interregionale migratie een verbeterde weergave van de werkelijkheid.

Het segmenteren van de bevolking naar *leeftijdsklassen* is -in verband met (a) beschikbare data en (b) gedragsrelevantie van de bevolking- de meest adequate manier om de interregionale migratie beter te benaderen. Dit kan gerealiseerd worden door een *cohort-model* van de bevolking te bouwen.

Mensen doorlopen een bepaalde levenscyclus, die bestaat uit meerdere levensfasen. De overgang van de ene levensfase naar de andere gaat vaak gepaard met een migratiewens. Niet iedereen doorloopt de verschillende levensfasen op dezelfde leeftijd, maar een algemene associatie kan wel gemaakt worden. Voor het bestuderen van migratiekrachten kan hiertoe een segmentatie van de bevolking gemaakt worden die correspondeert met verschillende levensfasen:

- a) *Kinderen*: 0-14 jaar:  
Migreren op grond van de migratiebeslissing van hun ouders. De migratiecoëfficiënt voor kinderen is dan ook sterk gecorreleerd met die van gezinnen (Rees e.a., 1998);
- b) *Scholieren en studenten*: 15-19 jaar:  
Onderwijsmogelijkheden spelen een belangrijke rol;
- c) *Arbeidsstarters*: 20-24 jaar:  
Arbeidsmarktfactoren in verband met toetreding in het arbeidsproces, de beschikbaarheid van woningen;
- d) *Settelaars*: 25-34 jaar:  
Arbeidsmarktfactoren en de beschikbaarheid van woningen relatief belangrijk, terwijl het aspect woonomgeving een rol gaat spelen;
- e) *Familliairen*: 35-49 jaar:  
Migratie relatief vaak op grond van woonomgevingsaspecten (opgroeïende kinderen), arbeidsmarktfactoren nemen in belang af;
- f) *Ervaren werkers*: 50-64 jaar:  
Migratie relatief vaak op grond van woonomgevingsaspecten (opgroeïende kinderen), arbeidsmarktfactoren nemen in belang af;
- g) *Futters*: 65-79 jaar:  
Arbeidsmarktfactoren verliezen hun relevantie, pensioen-migratie;
- h) *Bejaarden*: 80+ jaar:  
Voorzieningen worden belangrijk.

Terwijl pubers en jonge volwassenen drukke regio's opzoeken, migreren 'futters' liever naar rustige gebieden (zoals Drenthe, Gelderland en Zeeland). Migratie door bejaarden is sterk gecorreleerd aan de beschikbaarheid van voorzieningen. Alhoewel dergelijke voorzieningen overal worden aangetroffen, is er in sommige regio's sprake van een sterkere concentratie.

## Bijlage 7.4 Modelling reistijdontwikkeling

### *Definiëring van de causaliteit*

Het woon-werkverkeer kent een sterke mate van piekvorming, op allerlei plaatsen gepaard gaande met congestievorming. Het woon-werkverkeer is daarmee bepalend voor de structuur en dimensionering (en daarmee de kosten) van verkeersinfrastructuur, en bij het openbaar vervoer voor de structuur, frequentie en capaciteit van de aangeboden diensten. Bij planning van vervoerssystemen is inzicht in de ruimtelijke patronen van de woon-werkrelaties dan ook essentieel.

De ontwikkeling van interregionale infrastructuur/reistijden kan op basis de interregionale pendel endogeen worden verkend. Voor dit idee is in deze paragraaf een eerste opzet gegeven. Hierbij wordt verondersteld dat de *ontwikkeling* in interregionaal woon-werk verkeer op lange termijn zorgt een *ontwikkeling* van interregionale reistijden doordat in infrastructuur wordt geïnvesteerd. Impliciet wordt verondersteld dat de *ontwikkeling* in het goederenvervoer en werk-werk-relaties niet bijdragen in de *ontwikkeling* van ruimtelijke interactie (of in dezelfde mate als het woon-werk verkeer).

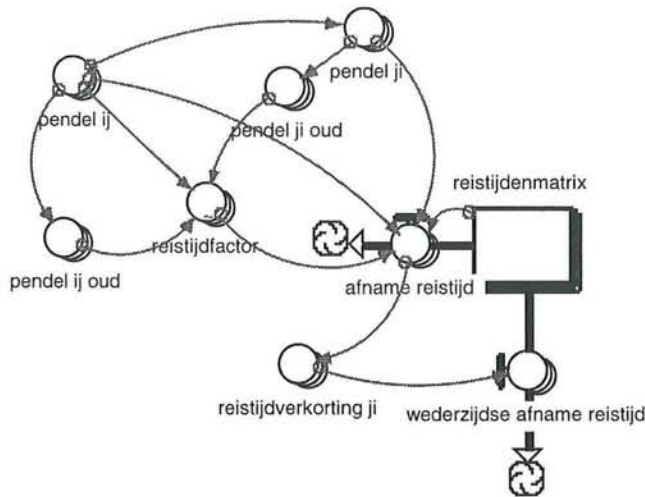
In een mogelijke modellering leidt een toename van pendel tussen  $i$  en  $j$  tot een toename van infrastructurele investeringen op die verbinding en een afname van reistijd op die verbinding. De ontwikkeling van die pendelstroom waardoor de totale, oude grootte van pendel wordt overtroffen, tussen twee regio's is hierbij *maatgevend* voor de uitbreidingsinvesteringen en de reistijdverkorting. Het effect van de reistijdverkorting geldt in beide richtingen tussen de twee regio's der beschouwing, indien wordt aangenomen dat de infrastructuurverbetering voor beide richtingen geldt.

Een afname van pendel leidt tot minder investeringen, maar hierdoor neemt de reistijd niet af indien verondersteld wordt dat de infrastructuur behouden blijft.

Doordat deze benadering in een zwaartekrachtmodel wordt toegepast, wordt rekening gehouden met netwerkrelaties. Stel dat in A veel werkgelegenheid ontstaat en dat hierdoor tussen regio A en B een reistijdverkorting ontstaat (vanwege congestie wordt er geïnvesteerd in infrastructuur). Hierdoor wordt ook een reistijdverkorting met andere (achterliggende) regio's C gerealiseerd. Dit moet zich in het model uiten doordat de toegenomen massa van A ook mensen uit C aantrekt. Is dit niet het geval, dan kan worden verondersteld dat de reistijdverkorting tussen A en C geen significante invloed heeft op deze interregionale reistijd. Dit rechtvaardigt de opzet van de voorgestelde modelstructuur.

Nadeel van deze benadering is dat in werkelijkheid reistijdverkortingen niet geleidelijk ontwikkelen (er wordt op één moment een infrastructuurverbetering gerealiseerd) en dat er vaak sprake is van overheidsingrijpen. De modellering kan worden verbeterd indien ook het goederenvervoer (bedrijfssectoren vereist) en werk-werk relaties in beeld worden gebracht. In dit geval wordt het subsysteem verkeer- en vervoer aan de interregionale modelstructuur toegevoegd. De reistijdontwikkeling kan verbeterd worden weergegeven door de absolute pendeltoename tegen de absolute, totale pendel uit te zetten.

### Stella-modellering reistijdontwikkeling



Figuur-Bijlage 7.13 Idee endogene reistijdontwikkeling

### Stella-listing reistijdontwikkeling

$reistijdenmatrix[van,naar](t) = reistijdenmatrix[van,naar](t - dt) + (- afname\_reistijd[van,naar] - wederzijdse\_afname\_reistijd[van,naar]) * dt$

INIT reistijdenmatrix[van,naar] = 1

afname\_reistijd[van,naar] = if pendel\_ij[van,naar]>pendel\_ji[naar,van] then

reistijdfactor[van,naar]\*reistijdenmatrix[van,naar] else 0

wederzijdse\_afname\_reistijd[van,naar] = reistijdverkorting\_ji[naar,van]

pendel\_ij[van,naar] = 0

pendel\_ij\_oud[van,naar] = DELAY(pendel\_ij[van,naar],1)

pendel\_ji[naar,van] = pendel\_ij[van,naar]

pendel\_ji\_oud[van,naar] = DELAY(pendel\_ji[naar,van],1)

reistijdverkorting\_ji[naar,van] = afname\_reistijd[van,naar]

reistijdfactor[van,naar] = GRAPH(if pendel\_ij\_oud[van,naar]>pendel\_ji\_oud[van,naar] then

pendel\_ij[van,naar]-pendel\_ij\_oud[van,naar] else pendel\_ij[van,naar]-pendel\_ji\_oud[van,naar])

(0.00, 0.00), (10000, 0.005), (20000, 0.005), (30000, 0.01), (40000, 0.025), (50000, 0.045),

(60000, 0.075), (70000, 0.11), (80000, 0.155), (90000, 0.21), (100000, 0.275)



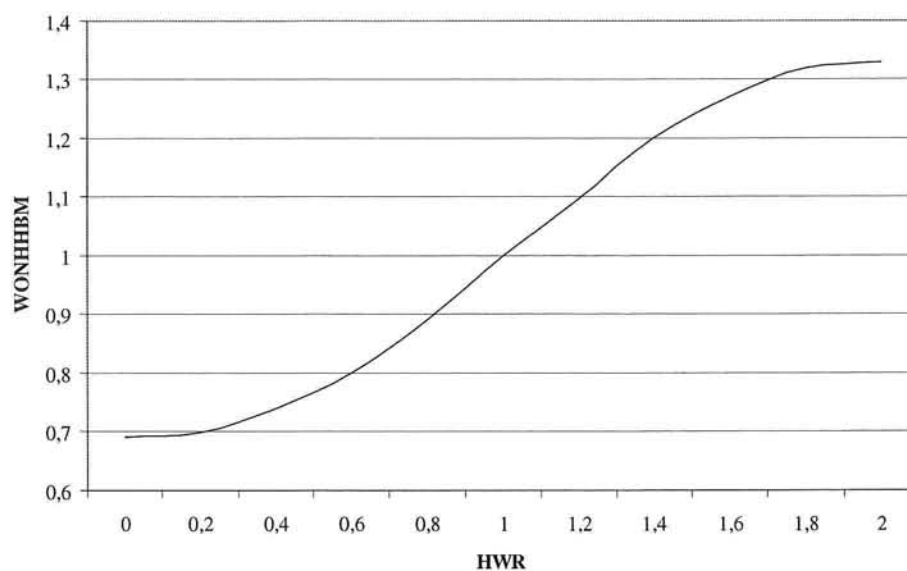
## **BIJLAGEN HOOFDSTUK 8**

### **MODULE WONINGMARKT**



## Bijlage 8.1 Multipliertafel module Woningmarkt

De nieuwbouw van woningen wordt aangestuurd met de multipliertafels *WONHHBM* (*woningen-huishoudens bouwmultiplier*). Het verloop van deze multipliertafel is weergegeven in figuur-Bijlage 8.1. Voor de theoretische achtergrond van de multipliertafel wordt verwezen naar paragraaf 8.3.1 in het eindrapport.



Figuur-Bijlage 8.1 Multipliertafel *WONHHBM*



## **BIJLAGEN HOOFDSTUK 9**

### **MODULE BEDRIJVEN**

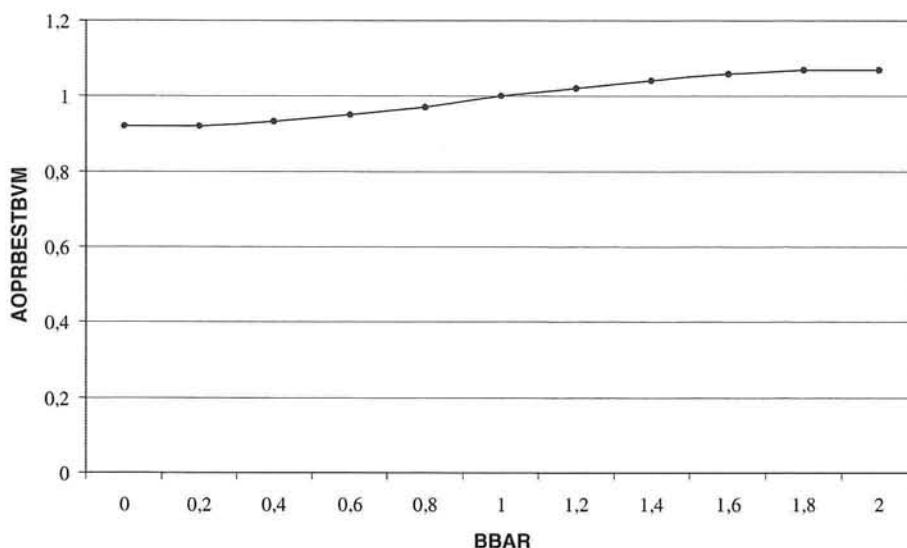


## Bijlage 9.1 Multipliertafels module Bedrijven

Deze bijlage geeft een overzicht van de gebruikte multipliertafels ten aanzien van het subsysteem bedrijven. De gepresenteerde multipliertafels zijn de feitelijke regulatoren in het model, welke zorgen voor de dynamiek tussen subsystemen. Hiermee sturen de multipliertafels de intraregionale en interregionale processen in het model aan. Voor de theoretische achtergrond van de multipliertafels wordt verwezen naar het bijbehorende hoofdstuk in het eindrapport.

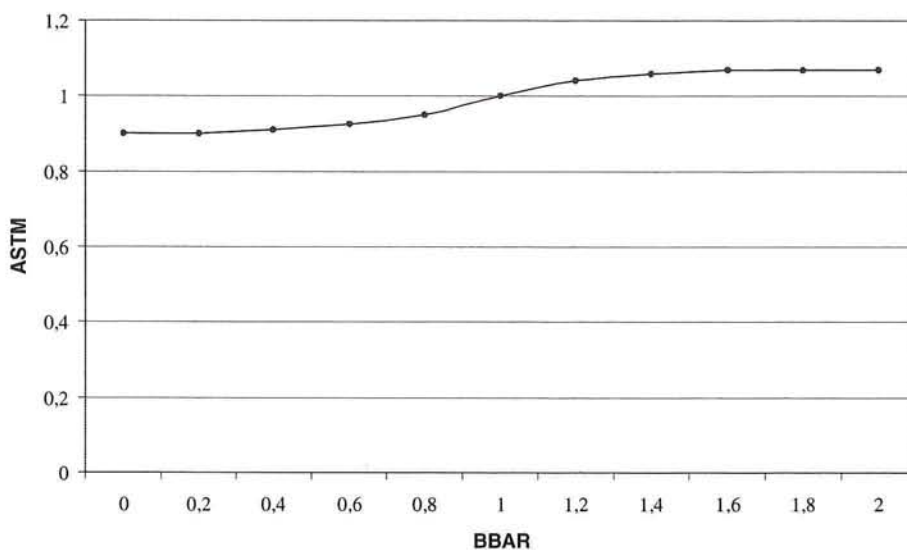
### Bijlage 9.1.1 Multipliertafels bedrijfsoprichting

De aansturing van bedrijfsoprichting geschiedt via twee multipliertafels. Het verloop van deze multipliertafels is weergegeven in figuur-Bijlage 9.1 en 9.2. De multiplier *AOPRBESTBVM* (*attractiviteit van bedrijfsoprichting vanuit bestaande bedrijven*) stuurt de oprichting van bedrijfsvestigingen vanuit bestaande bedrijven aan.



Figuur-Bijlage 9.1 Multipliertafel AOPRBESTBVM

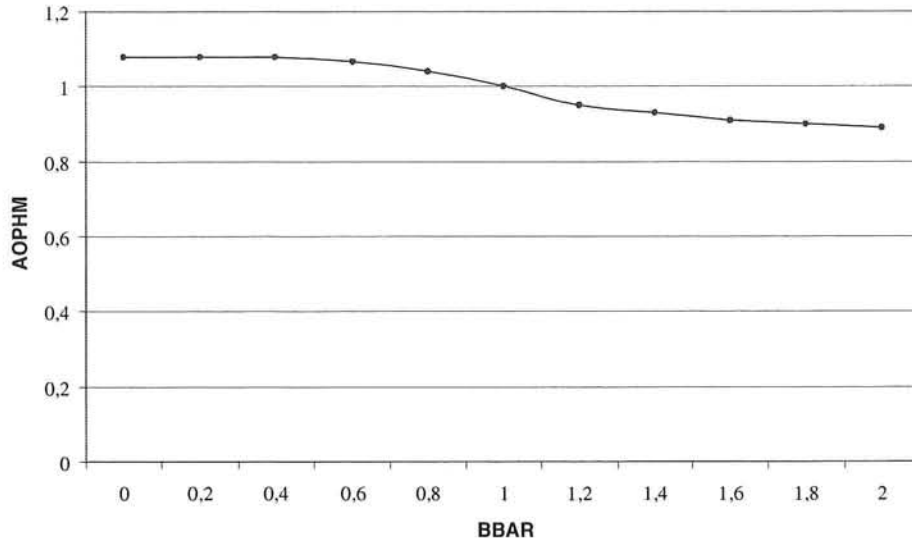
De multiplier *ASTM* (*attractiviteit van bedrijfsoprichting voor starters*) stuurt de oprichting van bedrijfsvestigingen door starters aan.



Figuur-Bijlage 9.2 Multipliertafel ASTM

### Bijlage 9.1.2 Multipliertafel bedrijfsopheffing

De aansturing van bedrijfsopheffing geschiedt via de multipliertafel. Het verloop van deze multipliertafels is weergegeven in figuur-Bijlage 9.3. De multiplier *AOPHM* ('attractiviteit' van *bedrijfsopheffing*) stuurt de opheffing van bedrijfsvestigingen aan.

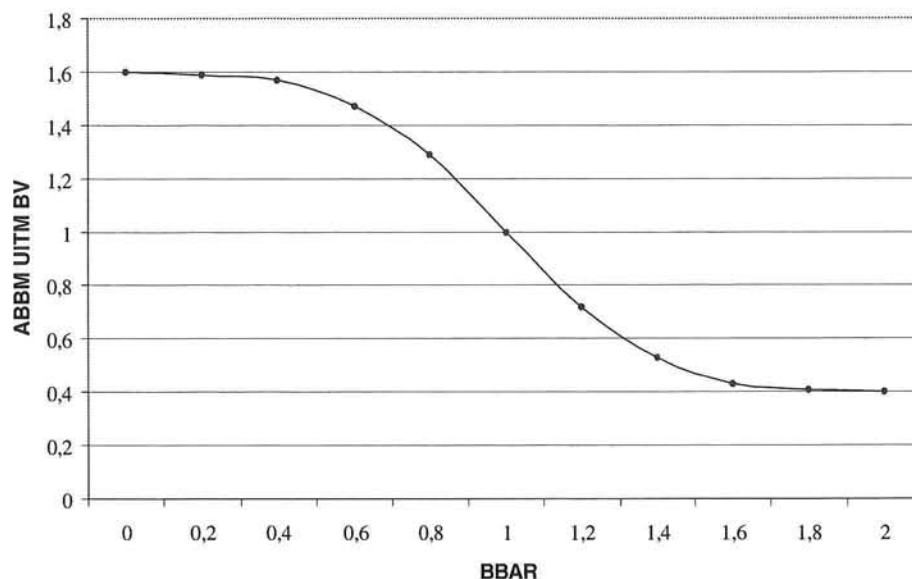


Figuur-Bijlage 9.3 Multipliertafel AOPHM

### Bijlage 9.1.3 Multipliertafels binnenlandse bedrijfsmigratie

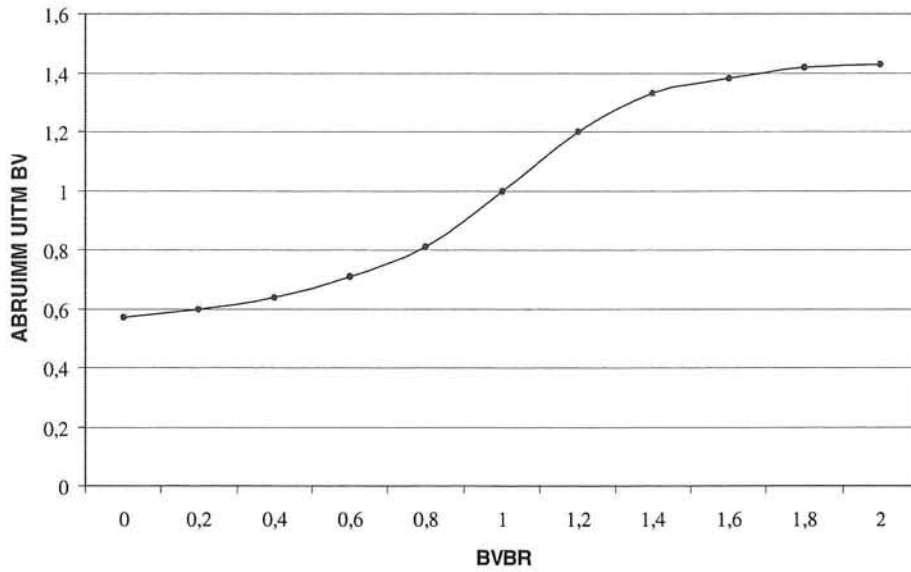
De aansturing van de potentiële uitmigratie van bedrijven geschiedt met twee multipliertafels. Dit zijn *ABBM\_UITM\_BV* (attractiviteit van beroepsbevolking voor potentiële uitmigratie van bedrijven) en *ABRUIMM\_UITM\_BV* (attractiviteit van bedrijfsruimte voor potentiële uitmigratie van bedrijven). Het verloop van deze multipliertafels is weergegeven in figuur-Bijlage 9.4 en in figuur-Bijlage 9.5.

De aansturing van de potentiële inmigratie van bedrijven geschiedt via de multipliertafels *ABBM\_INM\_BV* (attractiviteit van beroepsbevolking voor potentiële inmigratie van bedrijven) en *ABRUIMM\_INM\_BV* (attractiviteit van bedrijfsruimte voor potentiële inmigratie van bedrijven). Deze multipliertafels zijn weergegeven in figuur-Bijlage 9.6 en in figuur-Bijlage 9.7

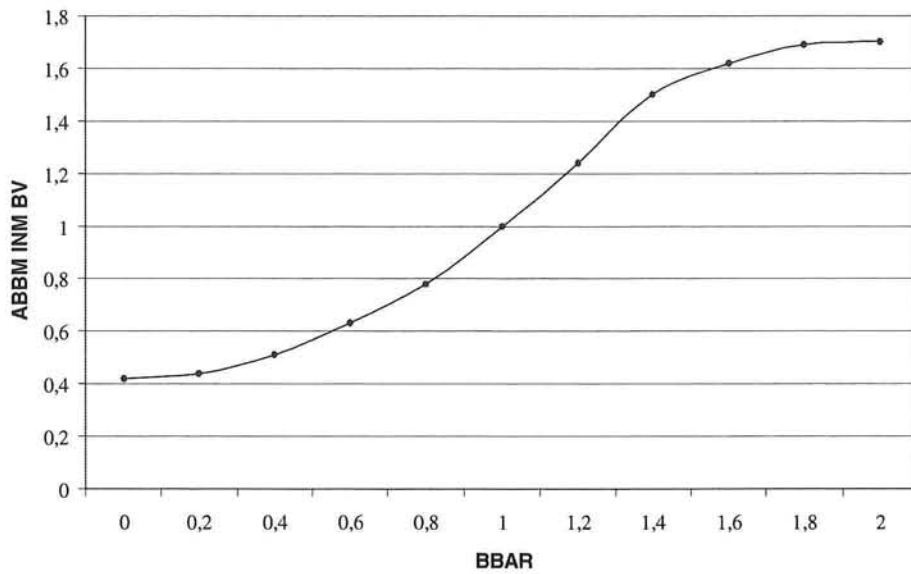


Figuur-Bijlage 9.4 Multiplier ABBM\_UITM\_BV

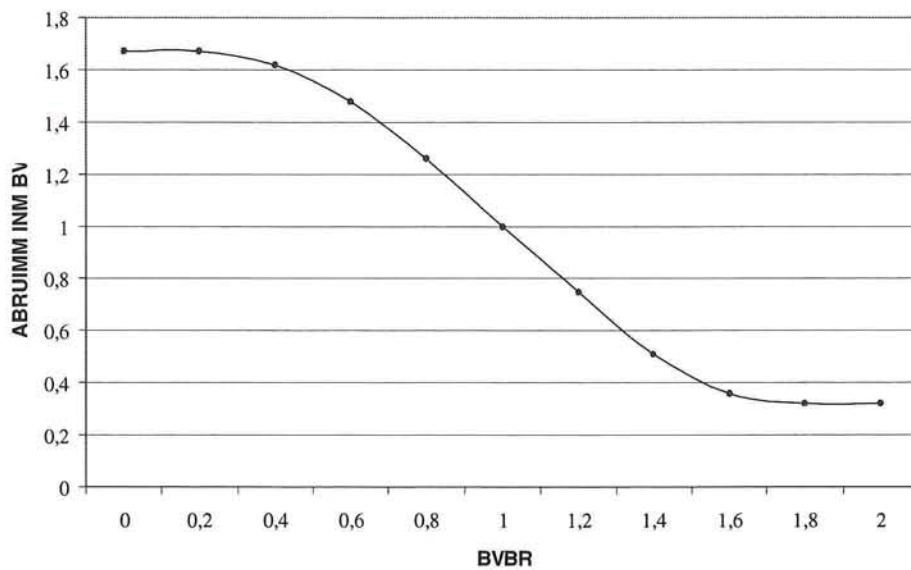




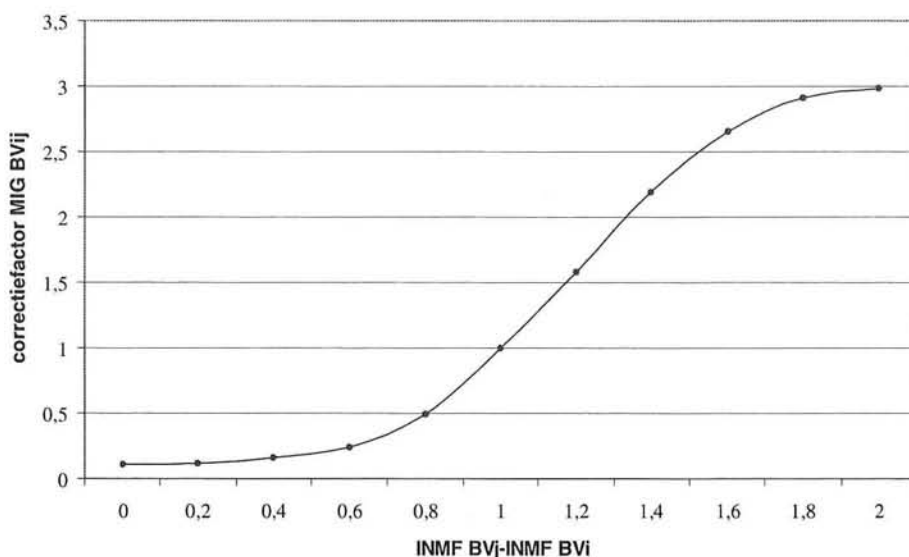
Figuur-Bijlage 9.5 Multipliertafel ABRUIMM\_UITM\_BV



Figuur-Bijlage 9.6 Multipliertafel ABBM\_INM\_BV



Figuur-Bijlage 9.7 Multipliertafel ABRUIMM\_INM\_BV



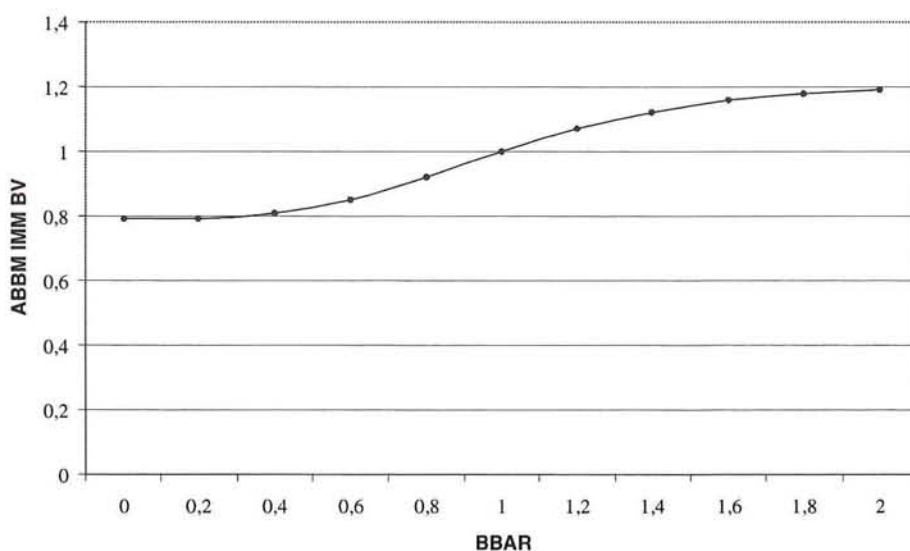
Figuur-Bijlage 9.8 Correctiefactor bedrijfsmigratie

De correctiefactor, welke het effect van verschil in arbeidsmarkt- en bedrijfsruimtemarkt-attractiviteit op de bedrijfsmigratie uitdrukt, is weergegeven in figuur-Bijlage 9.8.

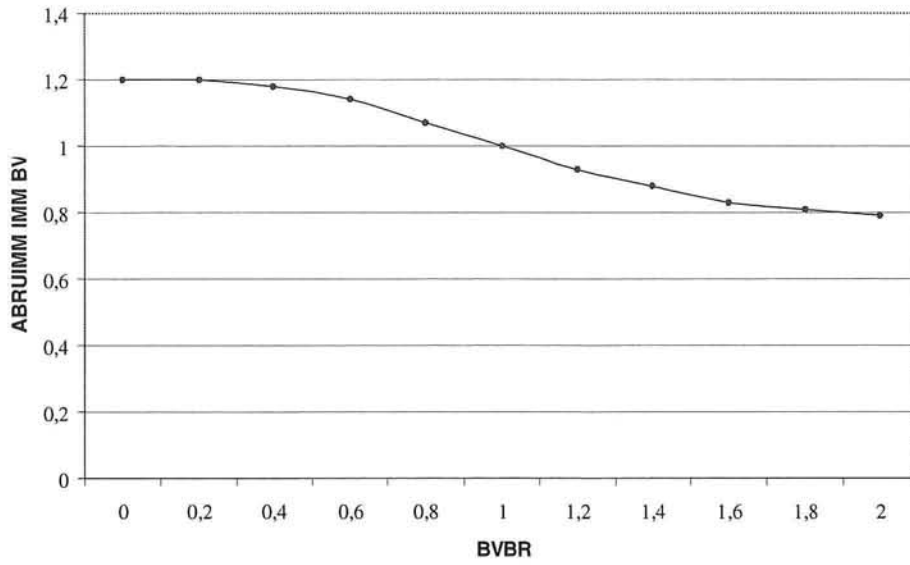
#### Bijlage 9.1.4 Multipliertafels buitenlandse bedrijfsmigratie

De aansturing van de immigratie van bedrijven geschiedt met twee multipliertafels. Dit zijn *ABBM\_IMM\_BV* (attractiviteit van beroepsbevolking voor immigratie van bedrijven) en *ABRUIMM\_IMM\_BV* (attractiviteit van bedrijfsruimte voor immigratie van bedrijven). Het verloop van deze multipliertafels is weergegeven in figuur-Bijlage 9.9 en in figuur-Bijlage 9.10.

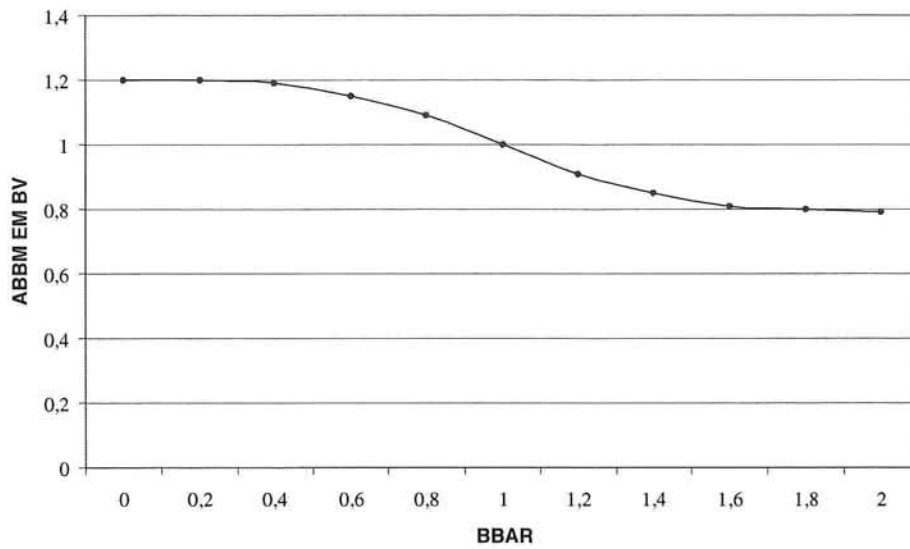
De aansturing van de emigratie van bedrijven geschiedt ook met twee multipliertafels. Dit zijn *ABBM\_EM\_BV* (attractiviteit van beroepsbevolking voor emigratie van bedrijven) en *ABRUIMM\_EM\_BV* (attractiviteit van bedrijfsruimte voor emigratie van bedrijven). Het verloop van deze multipliertafels is weergegeven in figuur-Bijlage 9.11 en in figuur-Bijlage 9.12.



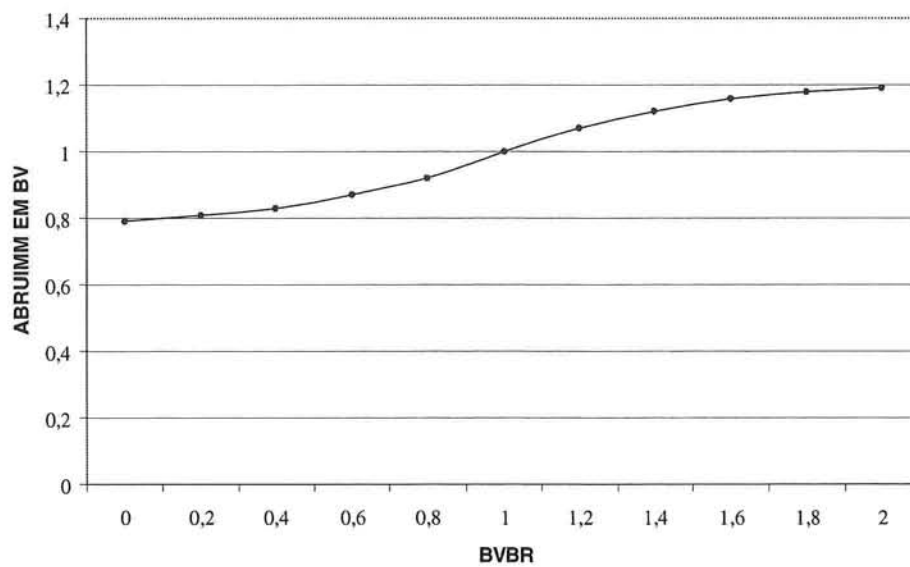
Figuur-Bijlage 9.9 Multipliertafel ABBM\_IMM\_BV



*Figuur-Bijlage 9.10 Multipliertafel ABRUIMM\_IMM\_BV*



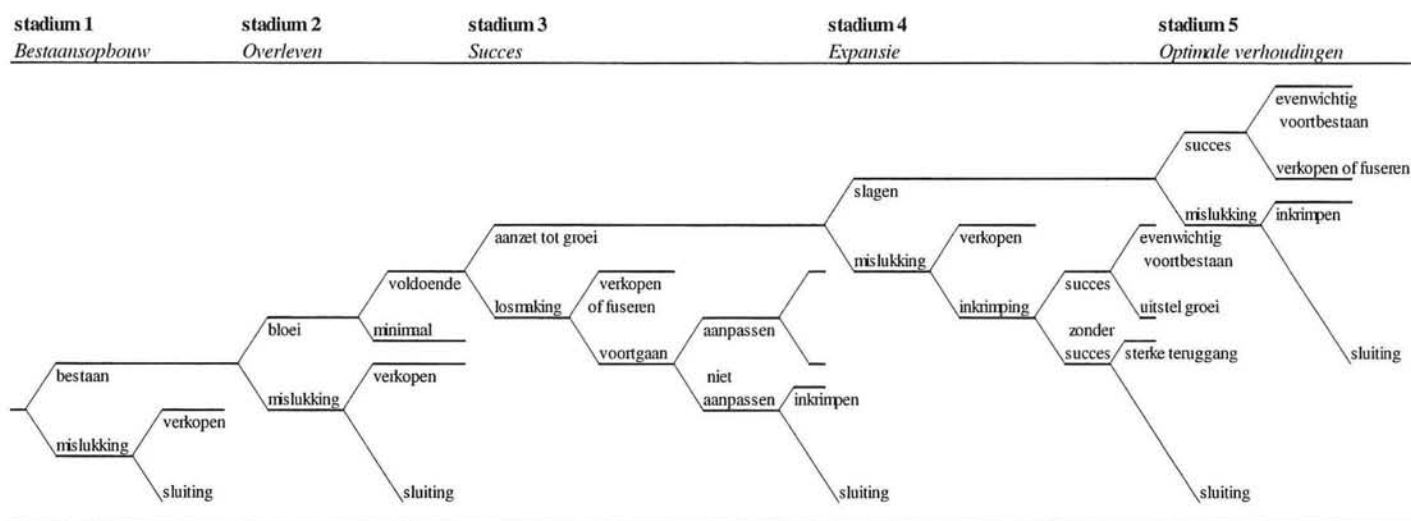
*Figuur-Bijlage 9.11 Multipliertafel ABBM\_EM\_BV*



*Figuur-Bijlage 9.12 Multipliertafel ABRUIMM\_EM\_BV*

## Bijlage 9.2 Levenscyclus van bedrijven

De levenscyclus van een bedrijf wordt gekenmerkt door 5 groeistadia: (1) *bestaansopbouw*, (2) *overleven*, (3) *succes*, (4) *expansie* en (5) *optimale verhoudingen*, zoals te zien is in figuur-Bijlage 9.13. Het verloop van de levenscyclus in de tijd is sterk bedrijfssectorspecifiek en is afhankelijk van de economische conjunctuur. De fase in de levenscyclus bepaald voor een groot gedeelte de bedrijfsvorming en -sluiting. Bedrijfssluiting en bedrijfsgroei kunnen in elke fase van de levensloop gerealiseerd worden.



Figuur-Bijlage 9.13 Levenscyclus van bedrijven

(bron: Van der Meer e.a., 1998)

De demografie van bedrijven wordt mede beïnvloed door fusies en overnames. Als bedrijven fuseren worden ‘overlappende’ vestigingen compact gereorganiseerd of opgeheven. In de huidige modellering wordt verondersteld dat dergelijke economische verschijnselen geen gevolgen hebben voor de vestiging van een bedrijf en het lokale productieproces. Aan de hand van bovenstaande cyclus kunnen dergelijke verschijnselen wel in beeld worden gebracht.

De grootte van een bedrijfsvestiging hangt nauw samen met de fase in de levenscyclus waar een bedrijf zich in bevindt. Ten aanzien van de *bedrijfs grootte* kan in eerste instantie onderscheid worden gemaakt in (1) het Kleinbedrijf, (2) het Middenbedrijf en (3) het Grootbedrijf.

Tot het middenbedrijf worden in Nederland alle particuliere ondernemingen gerekend, die een winsttoogmerk hebben en tussen 10 en 100 werknemers tellen. Bedrijven met minder dan 10 werknemers vormen het kleinbedrijf. Het midden- en kleinbedrijf (*MKB*) worden vaak samengevoegd en als geheel behandeld in de literatuur. Naast het MKB bestaat het grootbedrijf, waartoe particuliere ondernemingen met meer dan 100 werknemers behoren. Het MKB zorgt voor bijna 60% van de werkgelegenheid in het particuliere bedrijfsleven. Van het totaal aantal particuliere bedrijven in Nederland heeft het MKB een aandeel van bijna 99% (Van der Meer e.a., 1998).

### Bijlage 9.3 Segmentatie bedrijven

Voor een accurate beschrijving van dynamiek tussen economische activiteiten en overige subsystemen is onderscheid in verschillende bedrijfssectoren relevant. Ten eerste verschilt de vraag naar ruimte (naar aard en omvang) per type economische activiteit. Ten tweede kunnen kwalitatieve discrepanties op regionale arbeidsmarkten zorgen voor pendelstromen. Ten derde bestaan er tussen bedrijfssectoren bepaalde afzetrelaties, die mede bepalend zijn voor het optreden van congestie en de aantasting van het milieu.

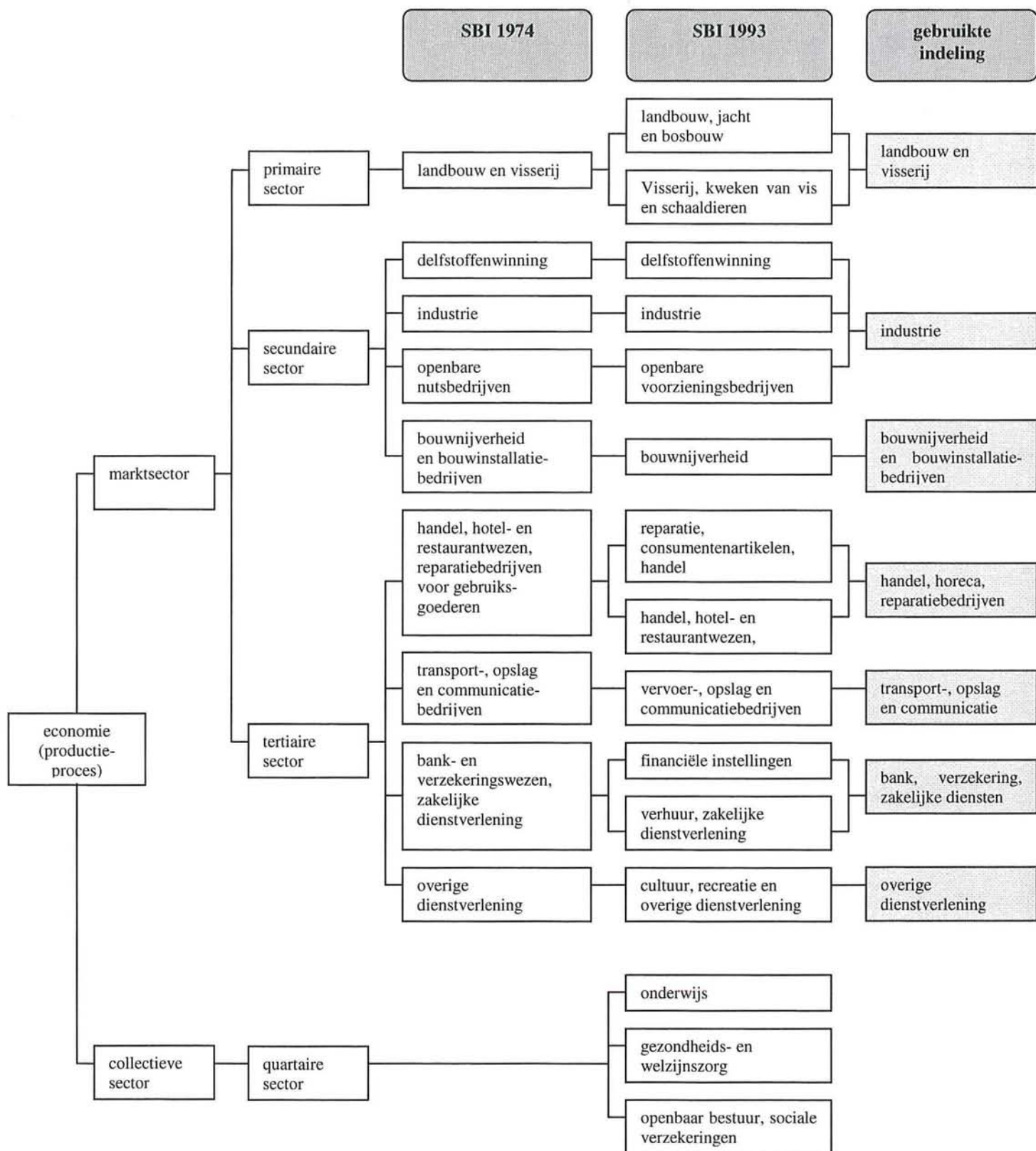
Een eerste segmentering is het onderscheid tussen *verzorgende* en *stuwende* activiteiten. Dit zijn twee relatieve begrippen die sterk gekoppeld zijn aan de keuze van resolutieniveau. *Stuwende bedrijven* zijn bedrijven waarvan de locatiekeuze niet in hoofdzaak wordt bepaald door het regionaal bevolkingsdraagvlak. De aanwezigheid van infrastructuur en een goede bereikbaarheid kunnen hierbij een belangrijke rol spelen (naast bijvoorbeeld aanwezigheid beroepsbevolking ertoe kan leiden dat daar stuwende bedrijvigheid komt). De bedrijfstakken 'industrie', 'handel', 'transport en communicatie' en 'bank- en verzekeringswezen en zakelijke dienstverlening' kunnen worden aangemerkt als stuwend vanwege het afwijkende ruimtelijke spreidingspatroon ten opzichte van de bevolking (Louter e.a., 1994).

*Verzorgende activiteiten* zijn op de *regionaal aanwezige* stuwende bedrijven én verzorgende bedrijven én bevolking gericht ('locale consumenten'). De groei van de verzorgende sector is dus direct afhankelijk van de ontwikkelingen in de stuwende sector (hetzij via leveringen aan stuwende bedrijven, hetzij via de consumptie van de werknemers in de stuwende sector) en de bevolking. Groei van stuwende sectoren zorgt zodoende voor directe werkgelegenheidsgroei, en indirect via groei van de verzorgende sector. Stuwende en verzorgende activiteiten kunnen verder gesegmenteerd worden naar verschillende bedrijfstakken.

Beperkt door databeschikbaarheid, kan voor een gedetailleerd model onderscheid worden gemaakt tussen 7 sectoren (figuur-Bijlage 9.14):

- a) Landbouw en visserij;
- b) Industrie (inclusief Delfstoffenwinning en Openbare nutsbedrijven);
- c) Bouwnijverheid en installatiebedrijven;
- d) Handel-, horeca- en reparatiebedrijven;
- e) Transport, opslag en communicatie;
- f) Bank, verzekering en zakelijke diensten;
- g) Overige dienstverlening.

Met betrekking tot aanwezige data classificeert het CBS economische activiteiten volgens de Standaard Bedrijfstakken Indeling (SBI). De SBI-indeling is in 1970, 1974 en 1993 veranderd; waarbij de bedrijfsklassen onderling niet volledig bij elkaar aansluiten. Met name vanaf 1993 is data op consistente basis bekend. Voor overige jaren dienen extrapolaties te worden gemaakt.



Figuur-Bijlage 9.14 Mogelijk onderscheid naar bedrijfssectoren

(bron: CB; Planologische Kengetallen)

## **BIJLAGEN HOOFDSTUK 10**

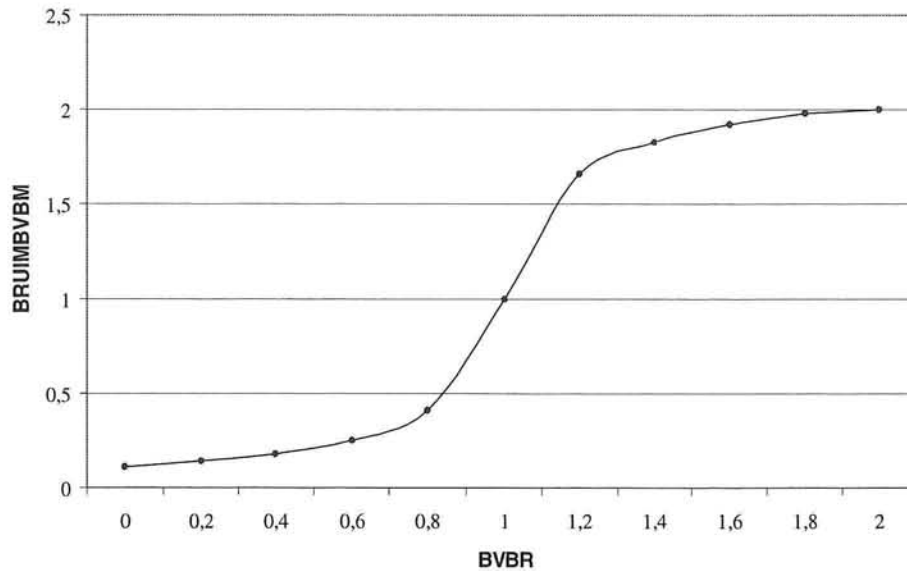
### **MODULE BEDRIJFSRUIMTEMARKT**





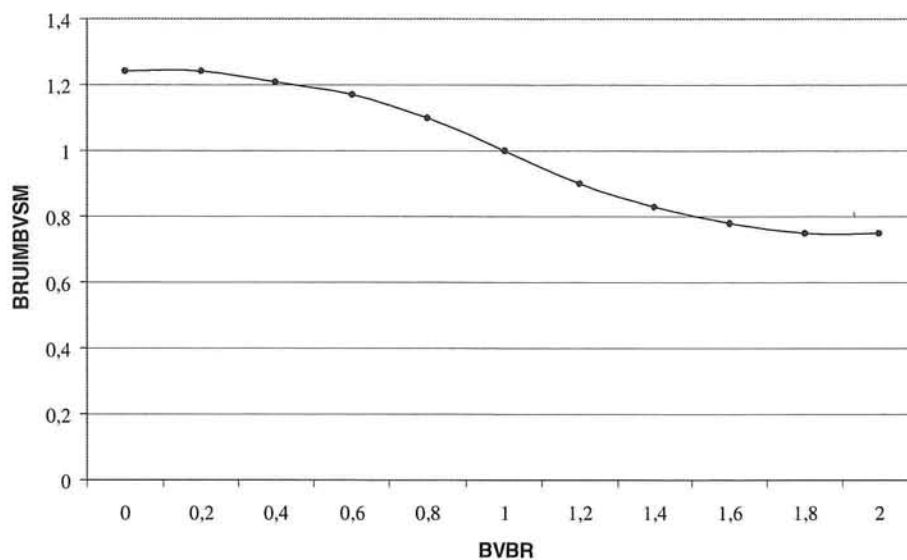
Deze bijlage geeft een overzicht van de gebruikte multipliertafels ter aansturing van het subsysteem bedrijfsruimten. Voor de theoretische achtergrond van de multipliertafels wordt verwezen naar het bijbehorende hoofdstuk in het eindrapport.

De multipliertafel *BRUIMBVBM* (*bedrijfsruimte-bedrijfsvestigingen bouwmultiplier*) stuurt de aanleg van bedrijfsruimten aan. Het verloop van deze multipliertafel is weergegeven in figuur-Bijlage 10.1.



Figuur-Bijlage 10.1 Multipliertafel *BRUIMBVBM*

De multipliertafel *BRUIMBVSM* (*bedrijfsruimte-bedrijfsvestigingen sloopmultiplier*) stuurt de sloop van bedrijfsruimten aan. Het verloop van deze multipliertafel is weergegeven in figuur-Bijlage 10.2.



Figuur-Bijlage 10.2 Multipliertafel *BRUIMBVSM*



## **BIJLAGEN HOOFDSTUK 12**

### **MODULE GRONDGEBRUIK**

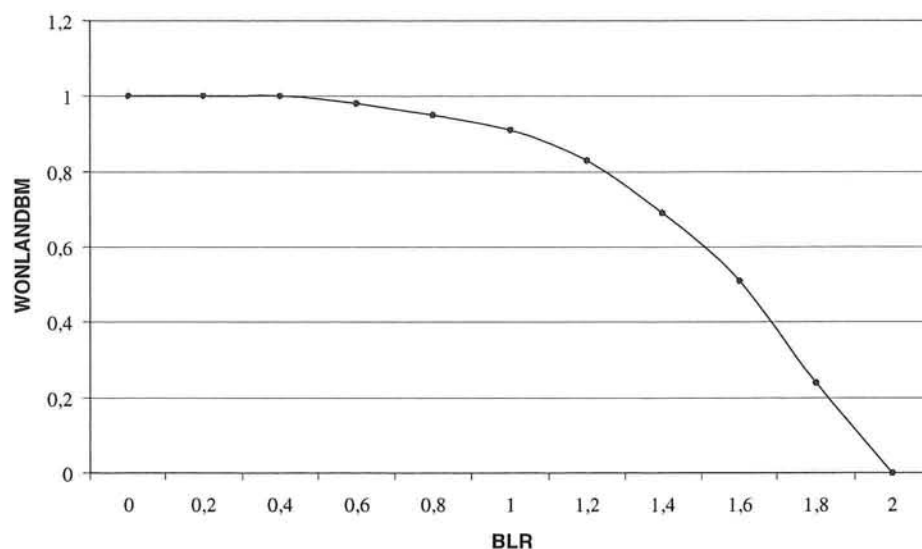


## Bijlage 12.1 Multipliertafels module Grondgebruik

Deze bijlage geeft een overzicht van de gebruikte multipliertafels ten aanzien van het subsysteem grondgebruik. De gepresenteerde multipliertafels zijn de feitelijke regulatoren in het model, welke zorgen voor de dynamiek tussen subsystemen. Voor de theoretische achtergrond van de multipliertafels wordt verwezen naar het bijbehorende hoofdstuk in het eindrapport.

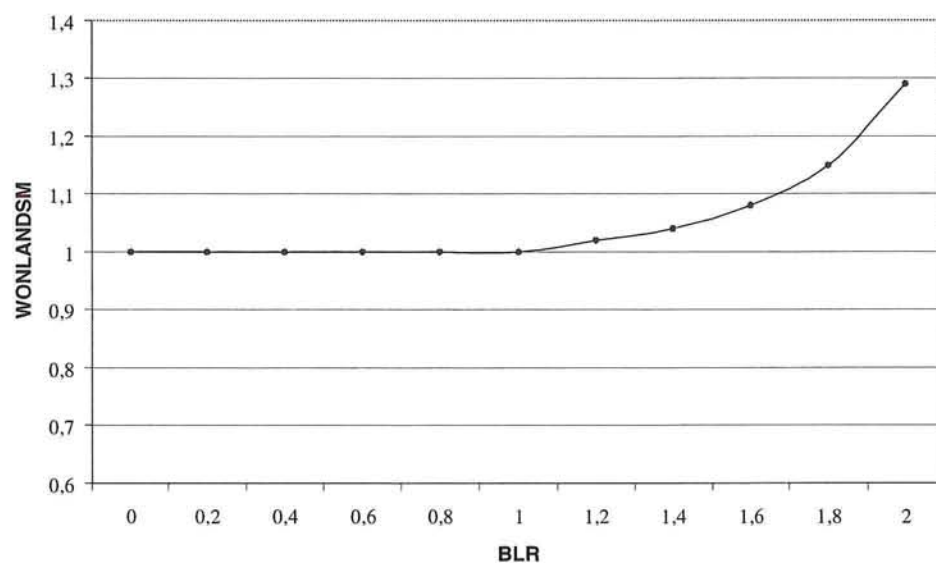
### Bijlage 12.1.1 Multipliertafels aansturing woningvoorraad

De aansturing van de woningbouw geschiedt via de multipliertafel *WONLANDBM* (*woningen-land bouwmultiplier*). Het verloop van deze multipliertafel is weergegeven in figuur-Bijlage 12.1.



Figuur-Bijlage 12.1 Multipliertafel *WONLANDBM*

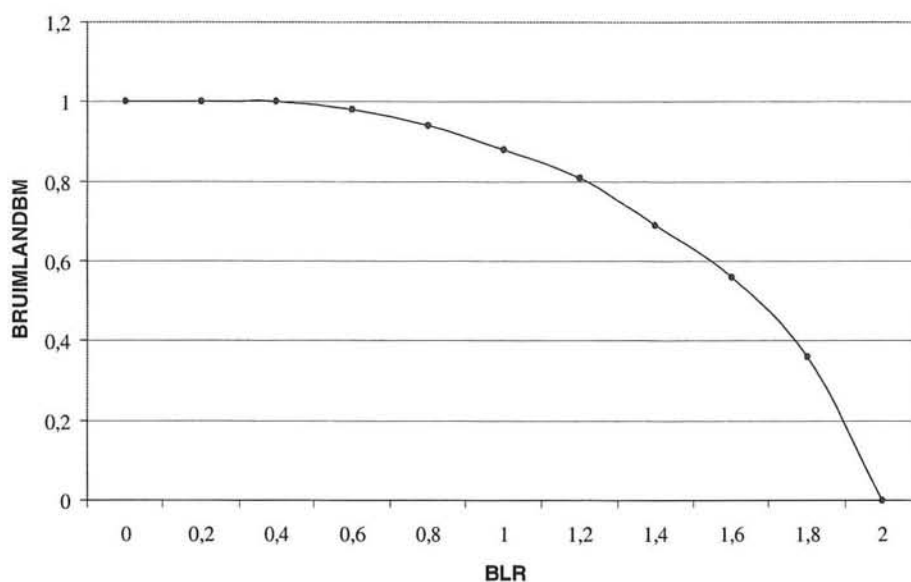
De sloop van woningen wordt aangestuurd met de multiplier *WONLANDSM* (*woningen-land sloopmultiplier*). Het verloop van deze multipliertafel is weergegeven in figuur-Bijlage 12.2.



Figuur-Bijlage 12.2 Multipliertafel *WONLANDSM*

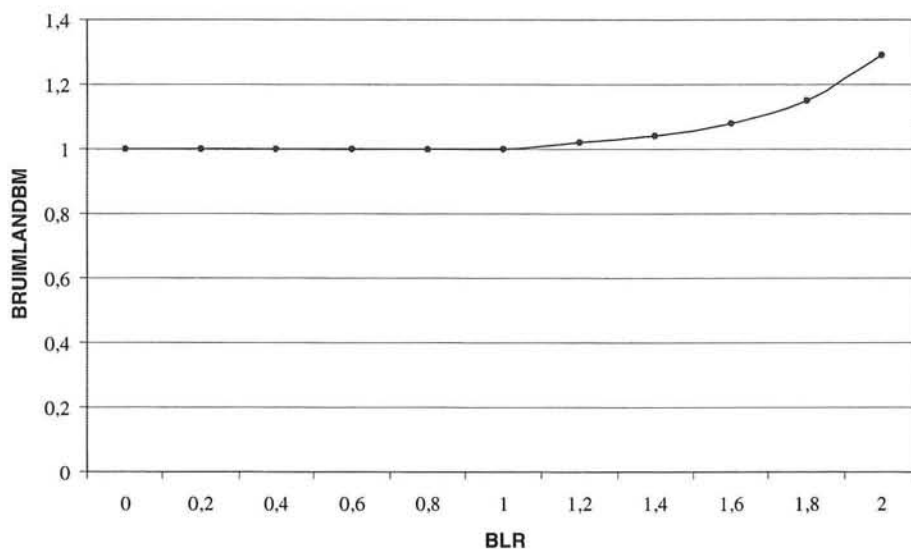
### Bijlage 12.1.2 Multipliertafels aansturing bedrijfsruimtevoorraad

De aansturing van de bouw van bedrijfsruimten geschiedt via de multipliertafel *BRUIMLANDBM* (*bedrijfsruimte-land bouwmultiplier*). Het verloop van deze multipliertafel is weergegeven in figuur-Bijlage 12.3.



Figuur-Bijlage 12.3 Multipliertafel *BRUIMLANDBM*

De aansturing van de sloop van bedrijfsruimten geschiedt via de multipliertafel *BRUIMLANDSM* (*bedrijfsruimte-land sloopmultiplier*). Het verloop van deze multipliertafel is weergegeven in figuur-Bijlage 12.4.



Figuur-Bijlage 12.4 Multipliertafel *BRUIMLANDSM*

## **BIJLAGEN HOOFDSTUK 13**

### **HINDCAST'S**



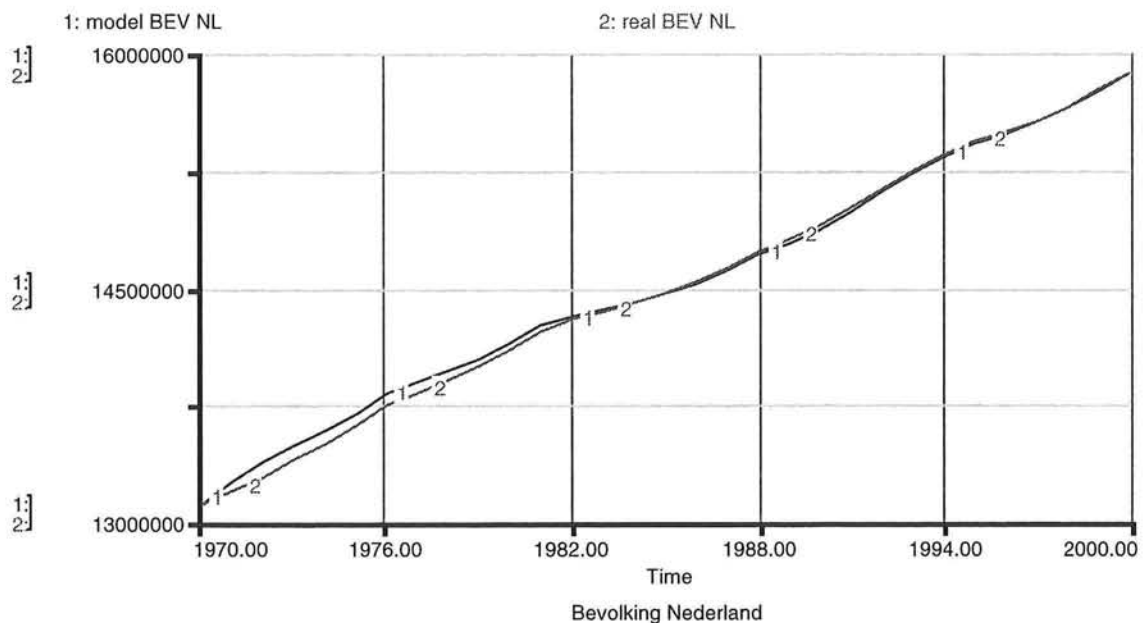


## Bijlage 13.1 Hindcast Nederland

Het modelresultaat op nationaal niveau geeft een indicatie van het 'overall' resultaat van het model. Deze paragraaf schetst een beeld van de hindcast van Nederland.

### Bijlage 13.1.1 Bevolking hindcast Nederland

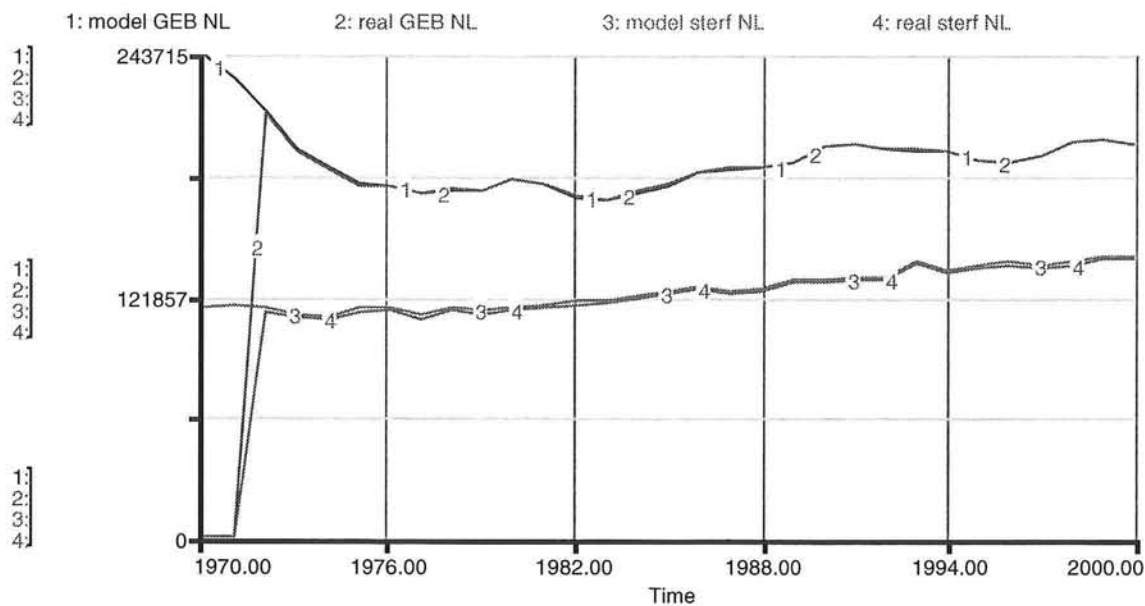
In figuur-Bijlage 13.1 is het modelresultaat van de Nederlandse bevolkingsontwikkeling uitgezet tegen de statistische tijdreeks van de Nederlandse bevolking. Hieruit kan geconcludeerd worden dat het model de ontwikkeling van de Nederlandse bevolking redelijk benadert.



Figuur-Bijlage 13.1 Hindcast bevolking Nederland

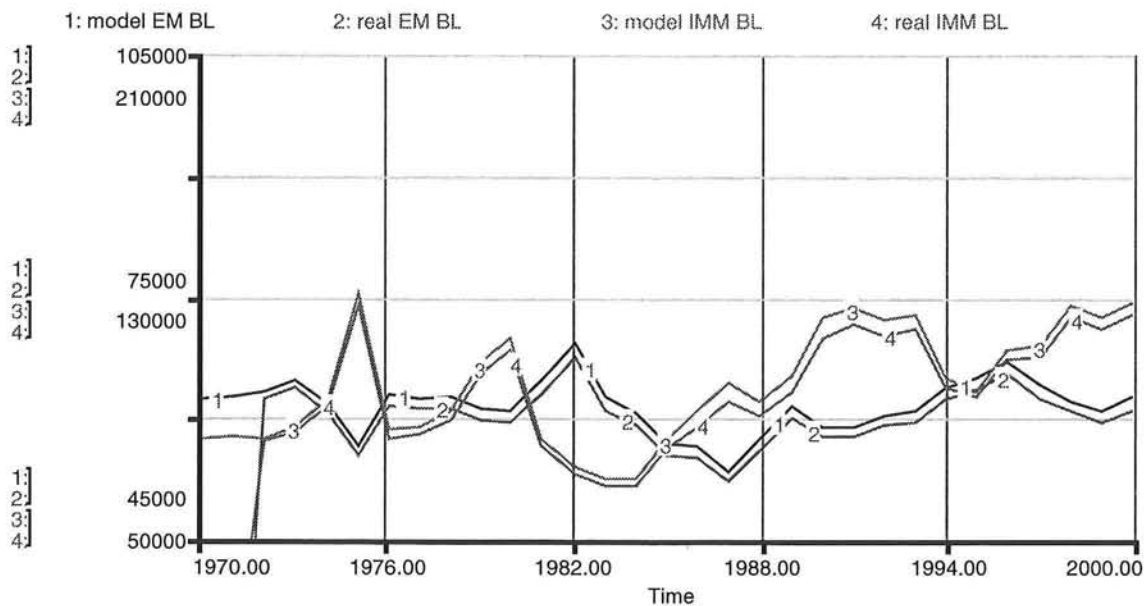
(bronstatistiek: CBS, ABF)

In het model is de bevolking een voorraadgrootte, welke accumuleert door natuurlijke aanwas (geboorte en sterfte), binnenlandse migratie (immigratie en uitmigratie) en buitenlandse migratie (immigratie en emigratie). Voor inzicht in de bevolkingsontwikkeling kunnen deze onderliggende stromen worden bekeken. In het geval van de bevolkingsontwikkeling op nationaal niveau geeft geboorte en sterfte (figuur-Bijlage 13.2), immigratie en emigratie (figuur-Bijlage 13.3) en de grootte van de binnenlandse migratie (figuur-Bijlage 13.4) inzicht in de afwijking ten opzichte van de historische tijdreeks van de totale bevolking.



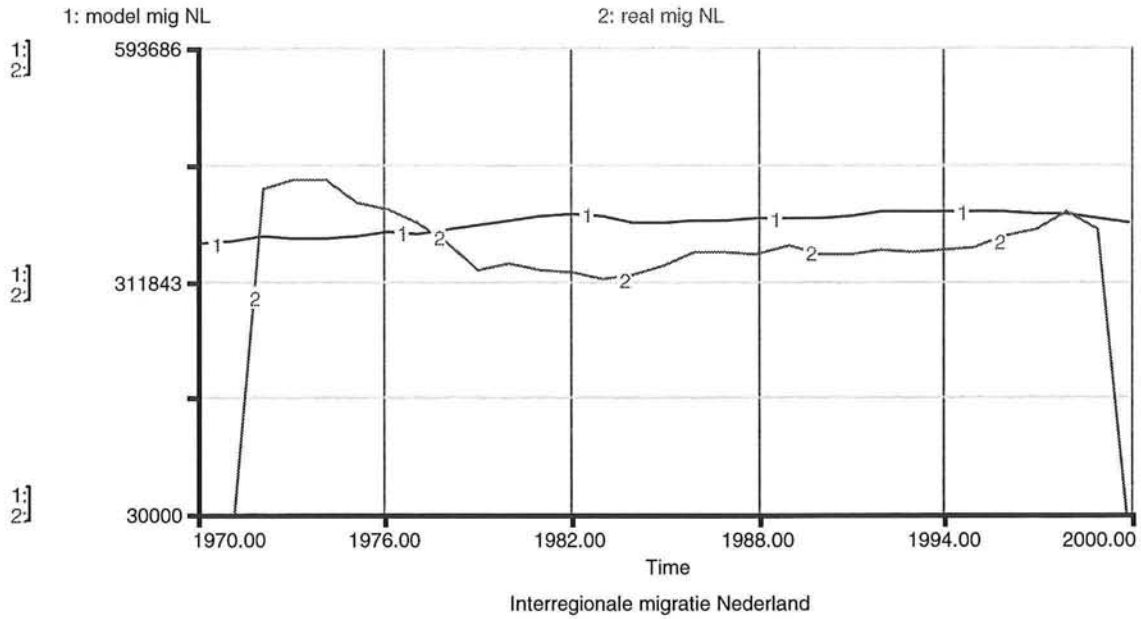
Natuurlijke aanwas Nederland

Figuur-Bijlage 13.2 Hindcast natuurlijke aanwas Nederland (bronstatistiek: CBS, ABF)



Buitenlandse migratie Nederland

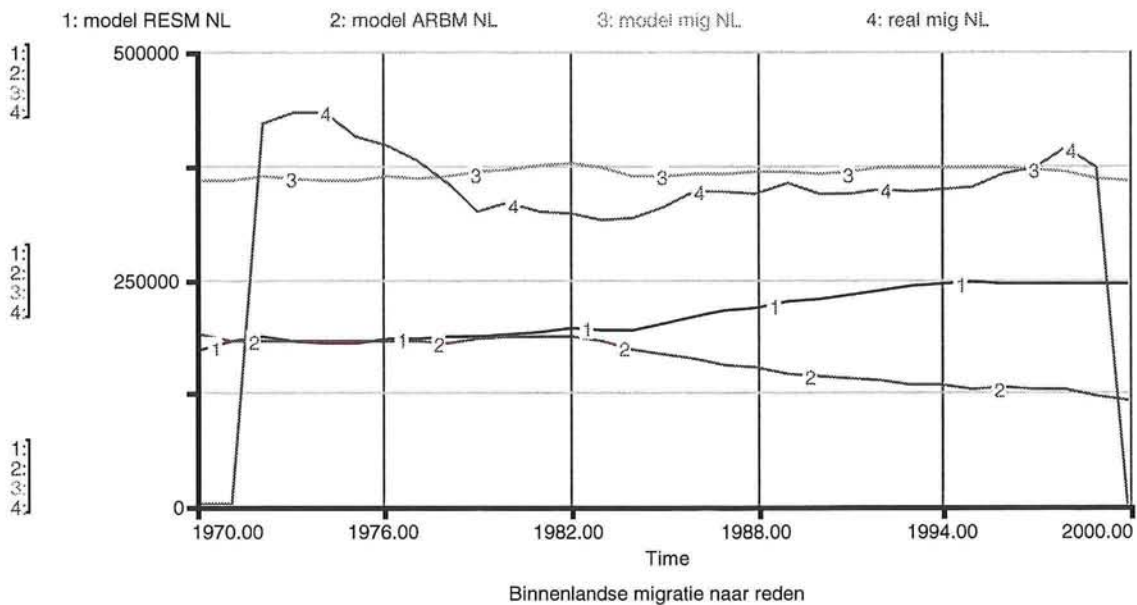
Figuur-Bijlage 13.3 Hindcast buitenlandse migratie Nederland (bronstatistiek: CBS, ABF)



Figuur-Bijlage 13.4 Hindcast binnenlandse migratie Nederland (bronstatistiek: CBS, ABF)

Zoals uit de modelresultaten blijkt, is de onnauwkeurigheid in de bevolkingsontwikkeling te wijten aan de onnauwkeurige weergave van de binnenlandse migratie. Door de onnauwkeurigheid in de binnenlandse migratie, wordt de weergave van de gekalibreerde natuurlijke aanwas en buitenlandse migratie verstoord. Voor inzicht in de binnenlandse migratie dient het Corop-niveau te worden bestudeerd. Dit is in het hoofdrapport aan de hand van de regio Utrecht geïllustreerd.

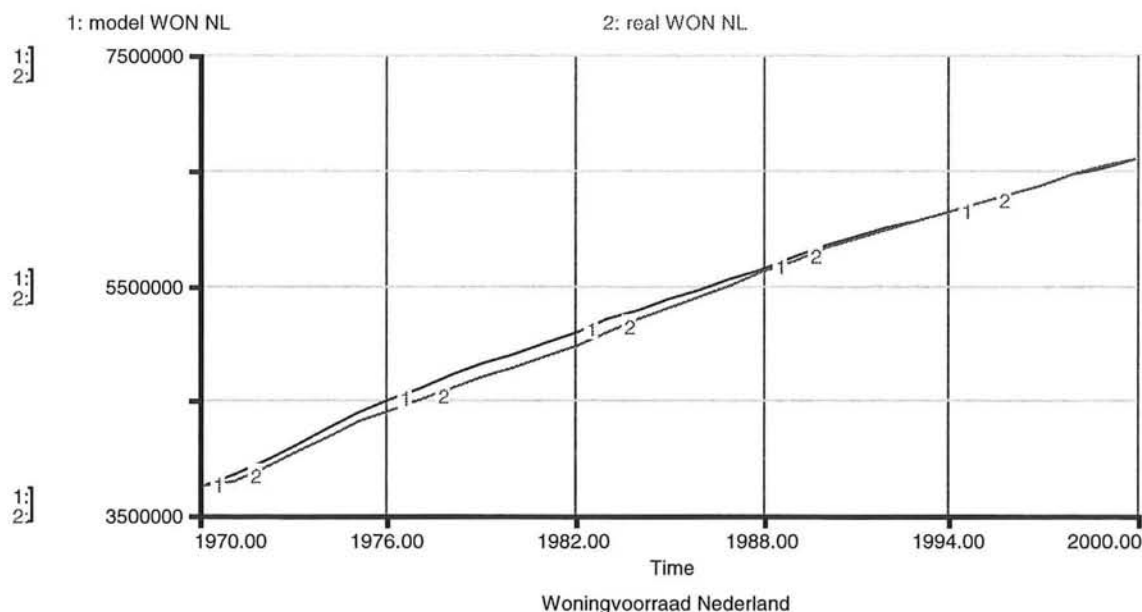
De grootte van de nationale migratie naar reden is weergegeven in figuur-Bijlage 13.5. Hieruit blijkt dat de arbeidsmigratie (ARBM) in belang is afgenomen, en residentiële migratie (RESM) in belang is toegenomen. De afname van arbeidsmigratie wordt in het model verklaard door de toegenomen bereidheid van mensen om te reizen.



Figuur-Bijlage 13.5 Hindcast binnenlandse migratie naar reden Nederland (bronstatistiek: CBS, ABF)

### Bijlage 13.1.2 Woningen hindcast Nederland

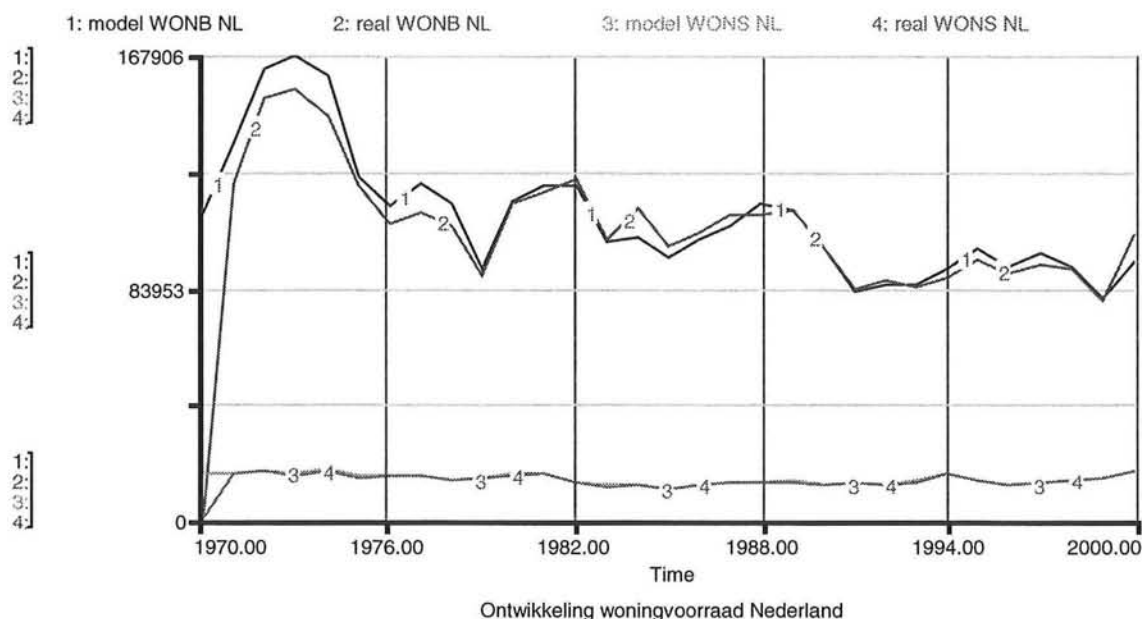
In figuur-Bijlage 13.6 is het modelresultaat van de ontwikkeling van de Nederlandse woningvoorraad uitgezet tegen de statistiek van de Nederlandse woningvoorraad. Hieruit kan geconcludeerd worden dat het model de ontwikkeling van de Nederlandse woningvoorraad redelijk benadert.



Figuur-Bijlage 13.6 Hindcast woningen Nederland

(bronstatistiek: CBS, ABF)

Voor meer inzicht in ontwikkelingen van de Nederlandse woningvoorraad kan beter naar het aantal nieuwbouwwoningen en onttrekkingen uit de woningvoorraad gekeken worden (figuur-Bijlage 13.7).



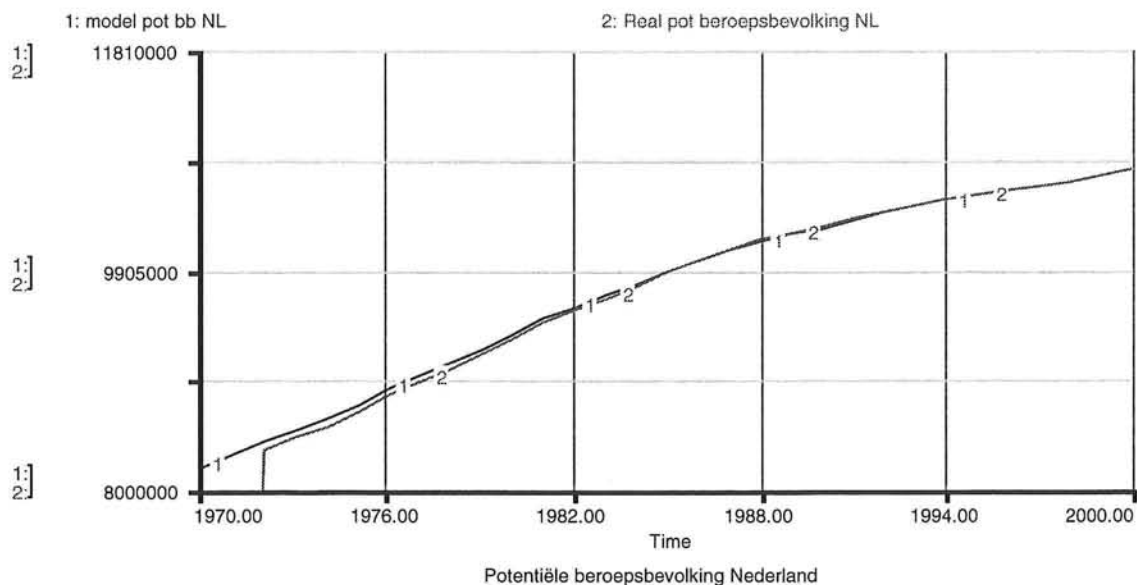
Figuur-Bijlage 13.7 Hindcast ontwikkelingen Nederlandse woningvoorraad (bronstatistiek: CBS, ABF)

De onnauwkeurigheid in de ontwikkeling van de Nederlandse woningvoorraad zit hem vooral in het aantal nieuwbouwwoningen. Dit kan deels verklaard worden door de waarde voor HHWR

(HuisHoudens-Woningen-Ratio). Door deze ratio is getracht de binnenlandse migratie op regionaal niveau in 'shape' te brengen.

### Bijlage 13.1.3 Arbeidsmarkt hindcast Nederland

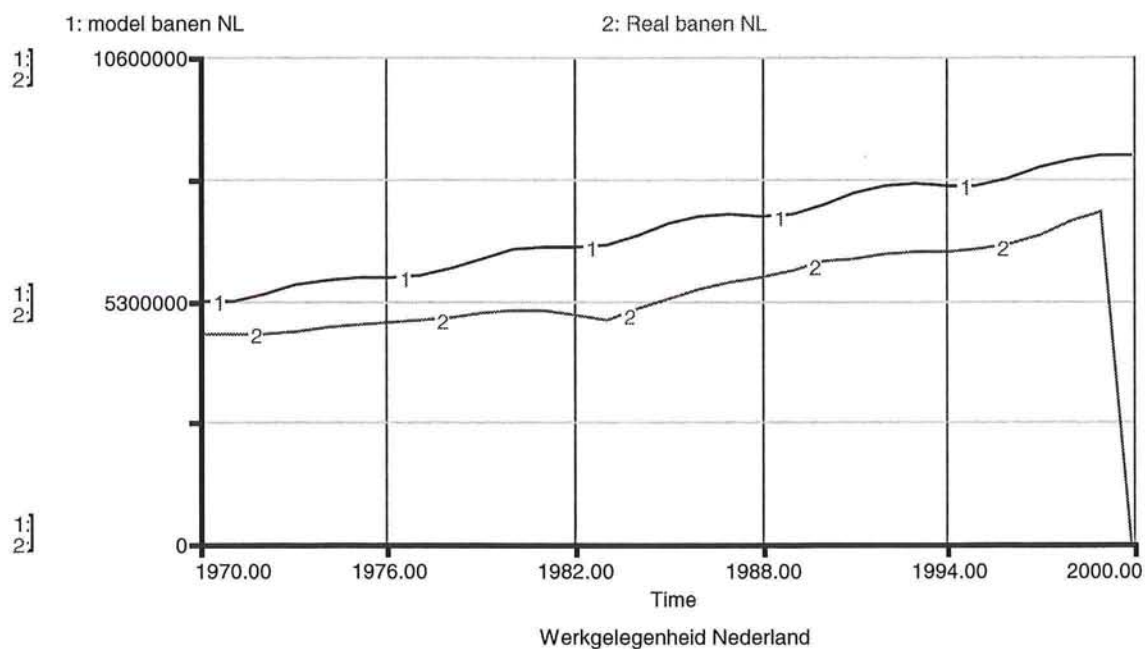
In figuur-Bijlage 13.8 en figuur-Bijlage 13.9 zijn de modelresultaten van de ontwikkeling van de Nederlandse arbeidsmarktomstandigheden uitgezet tegen historische tijdreeksen. Figuur 13.8 laat de potentiële beroepsbevolking (totale bevolking tussen het 15<sup>e</sup> en 64<sup>e</sup> levensjaar) zien.



Figuur-Bijlage 13.8 Hindcast potentiële beroepsbevolking Nederland (bronstatistiek: CBS, ABF)

Dit is in het model niet de tegenhanger van het aantal arbeidsplaatsen, aangezien slechts een gedeelte van de potentiële beroepsbevolking daadwerkelijk werkzaam of werkeloos is.

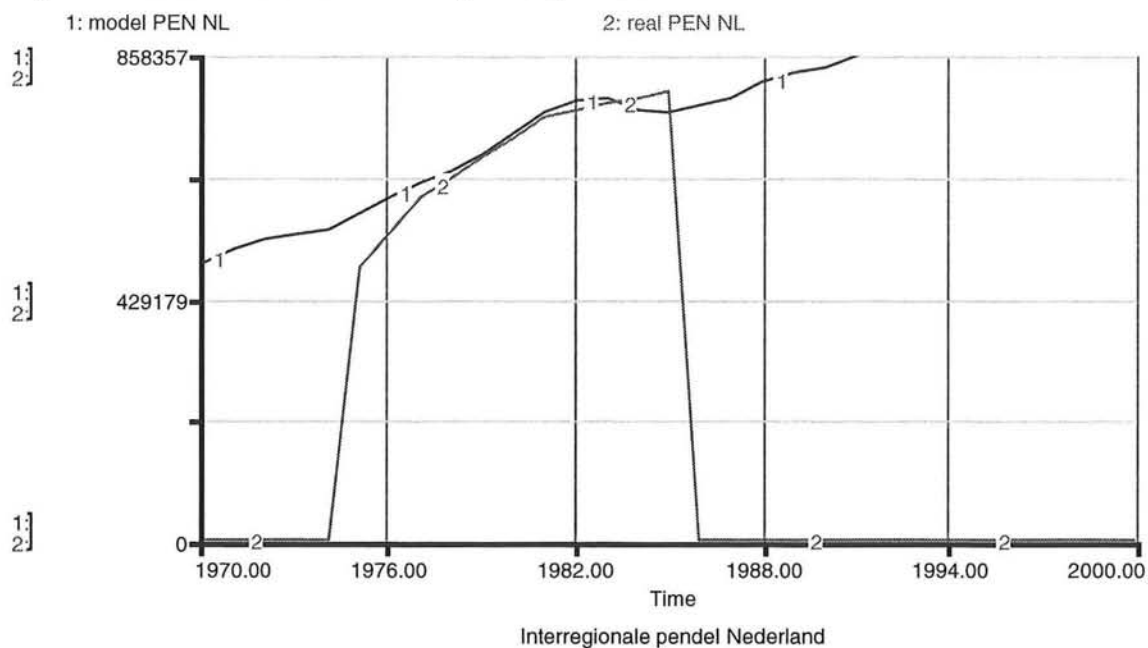
Figuur-Bijlage 13.9 laat de ontwikkeling van het aantal banen in Nederland zien.



Figuur-Bijlage 13.9 Hindcast arbeidsplaatsen Nederland (bronstatistiek: SWP/RARBON, ABF)

De historische tijdreeks is niet helemaal te vergelijken met het modelresultaat, aangezien deeltijd- en flexibele banen hier niet in zitten.

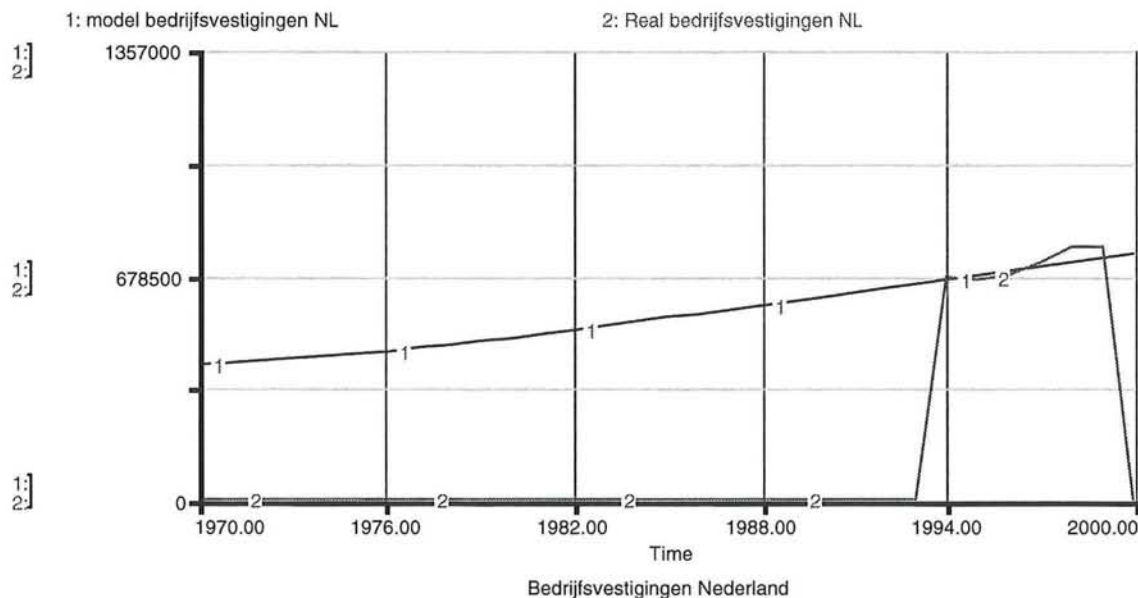
Het modelresultaat aangaande interregionale pendel is uiteengezet tegen een statistische tijdreeks die is afgeleid uit het OVG (OnderzoeksVerplaatsingsGedrag). Ook deze tijdreeks moet niet te letterlijk worden genomen (wat overigens nooit de bedoeling is), maar de orde van grootte geeft enige houvast ter kalibratie van interregionale pendel.



Figuur-Bijlage 13.10 Hindcast interregionale pendel Nederland (bronstatistiek: OVG, CBS)

### Bijlage 13.1.4 Bedrijven hindcast Nederland

In figuur-Bijlage 13.11 is het modelresultaat van de ontwikkeling van de Nederlandse bedrijfsvestigingenpopulatie uitgezet tegen een kleine historische tijdreeks.



Figuur-Bijlage 13.11 Hindcast bedrijfsvestigingen Nederland (bronstatistiek: CBS, ABF)

Zoals te zien is lijkt de structurele ontwikkeling van het aantal bedrijfsvestigingen in de orde van grootte redelijk goed. Wat opvalt is dat in de praktijk de bedrijfsvestigingenpopulatie, net zoals het aantal banen, fluctueert. In de praktijk is echter met name bedrijfsgroei en –krimp verantwoordelijk voor de toe- en afname van de werkgelegenheid. De schommelingen in de populatie van het aantal bedrijfsvestigingen komen met name tot stand door oprichtingen en faillissementen van *kleine* bedrijven. Het gemiddelde aantal mensen betrokken bij een oprichting of sterfte is ongeveer 1, terwijl het gemiddelde aantal mensen in een bedrijfsvestiging gemiddeld ongeveer 10 is. In het model is de afleiding van de werkgelegenheid van het aantal bedrijfsvestigingen dan ook problematisch. Vandaar dat geen fluctuatie in de populatie van het aantal bedrijfsvestigingen is gemodelleerd.

Gevolg hiervan is dat de ontwikkeling van bedrijfsomroerend niet goed in beeld wordt gebracht, en de ruimtelijke inertie (die in deze markt een grote rol speelt) in de modellering niet uit de verf komt.

## Bijlage 13.2 Modelfouten hindcast bevolking Utrecht

In bijlage 13.2 worden de fouten die het model in de hindcast van de regio Utrecht maakt ten aanzien van de *ontwikkeling van de bevolking* beschreven. Dit geeft inzicht in welke de mate het model de demografie van de bevolking accuraat benadert.

Tabel-Bijlage 13.1 geeft inzicht in de fout van het model ten aanzien van de voorraad 'bevolking' van de regio Utrecht.

*Tabel-Bijlage 13.1 Modelfouten hindcast bevolking Utrecht*

jaar	Bevolking model	Bevolking statistiek	Fout bevolking
1972	849.925	848.645	0,15%
1973	856.392	860.376	-0,46%
1974	862.935	872.328	-1,08%
1975	870.529	880.684	-1,15%
1976	880.219	890.991	-1,21%
1977	887.423	897.460	-1,12%
1978	894.826	902.787	-0,88%
1979	903.416	910.166	-0,74%
1980	913.849	920.199	-0,69%
1981	926.081	932.934	-0,73%
1982	935.002	942.304	-0,77%
1983	942.585	949.231	-0,70%
1984	950.465	955.995	-0,58%
1985	958.490	963.444	-0,51%
1986	967.406	972.268	-0,50%
1987	977.666	982.143	-0,46%
1988	988.801	993.973	-0,52%
1989	999.224	1.005.056	-0,58%
1990	1.009.500	1.015.942	-0,63%
1991	1.021.164	1.027.271	-0,59%
1992	1.032.728	1.037.727	-0,48%
1993	1.044.220	1.047.474	-0,31%
1994	1.055.126	1.056.477	-0,13%
1995	1.064.038	1.063.905	0,01%
1996	1.072.605	1.071.034	0,15%
1997	1.082.804	1.079.879	0,27%
1998	1.093.038	1.088.621	0,41%
1999	1.104.757	1.098.722	0,55%
<i>Gemiddelde absolute fout</i>			0,58%

(bronstatistiek: CBS, ABF)



Tabel-Bijlage 13.2 geeft inzicht in de fout van het model ten aanzien van de geboorte en sterfte in de Utrechtse bevolking.

Tabel-Bijlage 13.2 Modelfouten hindcast natuurlijke aanwas Utrecht

jaar	Geboorte model	Geboorte statistiek	Fout geboorte	Sterfte model	Sterfte statistiek	Fout sterfte
1972	13.998	13977	0,15%	7.244	7233	0,15%
1973	12.617	12676	-0,46%	6.789	6821	-0,46%
1974	11.855	11984	-1,08%	6.772	6846	-1,08%
1975	11.441	11574	-1,15%	7.209	7293	-1,15%
1976	11.351	11490	-1,21%	6.953	7038	-1,21%
1977	11.082	11207	-1,12%	6.782	6859	-1,12%
1978	11.309	11410	-0,88%	7.092	7155	-0,88%
1979	11.438	11523	-0,74%	6.940	6992	-0,74%
1980	11.870	11952	-0,69%	7.096	7145	-0,69%
1981	11.889	11977	-0,73%	7.239	7293	-0,73%
1982	11.485	11575	-0,77%	7.233	7289	-0,77%
1983	11.549	11630	-0,70%	7.263	7314	-0,70%
1984	12.046	12116	-0,58%	7.696	7741	-0,58%
1985	12.140	12203	-0,51%	7.557	7596	-0,51%
1986	12.935	13000	-0,50%	7.883	7923	-0,50%
1987	13.122	13182	-0,46%	7.898	7934	-0,46%
1988	13.200	13269	-0,52%	7.928	7969	-0,52%
1989	13.597	13676	-0,58%	8.260	8308	-0,58%
1990	14.064	14154	-0,63%	8.247	8300	-0,63%
1991	14.291	14376	-0,59%	8.386	8436	-0,59%
1992	14.305	14374	-0,48%	8.288	8328	-0,48%
1993	14.280	14324	-0,31%	8.816	8843	-0,31%
1994	14.130	14148	-0,13%	8.705	8716	-0,13%
1995	13.938	13936	0,01%	8.777	8776	0,01%
1996	14.214	14193	0,15%	9.037	9024	0,15%
1997	14.405	14366	0,27%	8.737	8713	0,27%
1998	15.091	15030	0,41%	9.006	8970	0,41%
1999	15.454	15370	0,55%	9.219	9169	0,55%
<i>Gemiddelde absolute fout</i>			0,58%	<i>Gemiddelde absolute fout</i>		0,58%

(bronstatistiek: CBS, ABF)

Tabel-Bijlage 13.3 geeft inzicht in de fout van het model ten aanzien van de binnenlandse immigratie en binnenlandse uitmigratie in de Utrechtse bevolking.

Tabel-Bijlage 13.3 Modelfouten hindcast binnenlandse migratie Utrecht

jaar	Binnenlandse immigratie model	Binnenlandse immigratie statistiek	Fout binnenlandse immigratie	Binnenlandse uitmigratie model	Binnenlandse uitmigratie statistiek	Fout binnenlandse uitmigratie	
1972	21.553	33923	-36,46%	22.698	29658	-23,47%	
1973	21.634	35570	-39,18%	21.965	30315	-27,54%	
1974	21.929	33675	-34,88%	21.521	32532	-33,85%	
1975	22.509	31819	-29,26%	21.486	29995	-28,37%	
1976	23.277	29850	-22,02%	21.921	29134	-24,76%	
1977	23.562	28253	-16,60%	21.830	28385	-23,09%	
1978	24.068	26775	-10,11%	21.989	25729	-14,54%	
1979	24.429	24906	-1,91%	22.006	22783	-3,41%	
1980	25.075	26176	-4,21%	22.444	22948	-2,20%	
1981	25.640	25271	1,46%	22.804	21895	4,15%	
1982	26.207	24808	5,64%	23.230	22266	4,33%	
1983	26.186	23698	10,50%	23.048	21455	7,43%	
1984	25.991	24055	8,05%	22.970	21289	7,90%	
1985	26.355	25567	3,08%	23.376	22635	3,27%	
1986	26.988	27174	-0,68%	24.046	24263	-0,90%	
1987	27.602	27926	-1,16%	24.855	24415	1,80%	
1988	28.158	27594	2,04%	25.331	23982	5,63%	
1989	28.611	28961	-1,21%	26.157	25254	3,58%	
1990	28.779	27694	3,92%	26.446	25111	5,32%	
1991	29.226	27140	7,69%	26.977	25504	5,77%	
1992	29.853	26807	11,36%	27.659	25720	7,54%	
1993	30.361	27022	12,36%	28.344	26123	8,50%	
1994	30.462	27182	12,07%	28.377	26147	8,53%	
1995	30.702	27384	12,11%	28.834	25987	10,96%	
1996	30.715	28718	6,95%	28.582	27088	5,52%	
1997	30.722	29063	5,71%	28.549	27534	3,68%	
1998	30.743	30910	-0,54%	28.573	29440	-2,94%	
1999	30.801	28625	7,60%	28.725	28035	2,46%	
<i>Gemiddelde absolute fout</i>			11,03%	<i>Gemiddelde absolute fout</i>			10,05%

(bronstatistiek: CBS, ABF)

Tabel-Bijlage 13.4 geeft inzicht in de fout van het model ten aanzien van de immigratie en emigratie in de Utrechtse bevolking.

Tabel-Bijlage 13.4 Modelfouten hindcast buitenlandse migratie Utrecht

jaar	Immigratie model	Immigratie statistiek	Fout immigratie	Emigratie model	Emigratie statistiek	Fout emigratie
1972	5.397	5389	0,15%	4.540	4533	0,15%
1973	5.477	5502	-0,46%	4.430	4451	-0,46%
1974	5.934	5999	-1,08%	3.830	3872	-1,08%
1975	7.961	8054	-1,15%	3.526	3567	-1,15%
1976	5.672	5741	-1,21%	4.222	4274	-1,21%
1977	5.760	5825	-1,12%	4.387	4437	-1,12%
1978	6.725	6785	-0,88%	4.433	4472	-0,88%
1979	7.595	7652	-0,74%	4.083	4113	-0,74%
1980	8.906	8968	-0,69%	4.079	4107	-0,69%
1981	6.082	6127	-0,73%	4.647	4681	-0,73%
1982	5.248	5289	-0,77%	4.895	4933	-0,77%
1983	4.768	4802	-0,70%	4.313	4343	-0,70%
1984	4.927	4956	-0,58%	4.273	4298	-0,58%
1985	5.357	5385	-0,51%	4.003	4024	-0,51%
1986	5.913	5943	-0,50%	3.648	3666	-0,50%
1987	6.633	6663	-0,46%	3.468	3484	-0,46%
1988	6.052	6084	-0,52%	3.729	3748	-0,52%
1989	6.563	6601	-0,58%	4.077	4101	-0,58%
1990	7.457	7505	-0,63%	3.942	3967	-0,63%
1991	7.326	7370	-0,59%	3.917	3940	-0,59%
1992	7.179	7214	-0,48%	3.898	3917	-0,48%
1993	7.406	7429	-0,31%	3.981	3993	-0,31%
1994	5.787	5794	-0,13%	4.385	4391	-0,13%
1995	5.923	5922	0,01%	4.384	4383	0,01%
1996	7.293	7282	0,15%	4.403	4397	0,15%
1997	6.731	6713	0,27%	4.339	4327	0,27%
1998	7.827	7795	0,41%	4.363	4345	0,41%
1999	7.510	7469	0,55%	4.261	4238	0,54%
<i>Gemiddelde absolute fout</i>			0,58%	<i>Gemiddelde absolute fout</i>		0,58%

(bronstatistiek: CBS, ABF)

### Bijlage 13.3 Modelfouten hindcast woningmarkt Utrecht

In bijlage 13.2 worden de fouten die het model in de hindcast van de regio Utrecht maakt ten aanzien van de *ontwikkeling van de woningmarkt* beschreven. Dit geeft inzicht in welke de mate het model de ontwikkeling van de woningvoorraad accuraat benadert.

Tabel-Bijlage 13.5 geeft inzicht in de fout van het model ten aanzien van de woningvoorraad van de regio Utrecht.

Tabel-Bijlage 13.5 Modelfouten hindcast woningvoorraad Utrecht

jaar	Woningvoorraad model	Woningvoorraad statistiek	Fout woningvoorraad
1972	247.023	240.343	2,78%
1973	257.198	248.848	3,36%
1974	267.494	258.363	3,53%
1975	277.181	267.585	3,59%
1976	285.235	274.367	3,96%
1977	292.637	280.977	4,15%
1978	300.572	287.809	4,43%
1979	307.943	296.130	3,99%
1980	313.615	301.380	4,06%
1981	321.223	310.996	3,29%
1982	329.171	317.963	3,52%
1983	337.408	326.153	3,45%
1984	344.177	334.562	2,87%
1985	351.071	343.767	2,12%
1986	357.439	350.718	1,92%
1987	364.397	358.570	1,63%
1988	371.646	366.532	1,40%
1989	379.447	375.565	1,03%
1990	387.166	384.451	0,71%
1991	393.755	392.114	0,42%
1992	399.394	399.182	0,05%
1993	405.327	405.338	0,00%
1994	410.987	410.597	0,09%
1995	417.139	416.045	0,26%
1996	423.874	421.558	0,55%
1997	430.166	428.648	0,35%
1998	436.312	434.710	0,37%
1999	442.668	440.409	0,51%
Gemiddelde absolute fout			2,09%

(bronstatistiek: CBS, ABF)

Tabel-Bijlage 13.6 geeft inzicht in de fout van het model ten aanzien van woningbouw en woningsloop in de Utrechtse woningvoorraad.

Tabel-Bijlage 13.6 Modelfouten hindcast ontwikkeling woningvoorraad Utrecht

jaar	Woningbouw model	Woningbouw statistiek	Fout woningbouw	Woningsloop model	Woningsloop statistiek	Fout woningsloop
1972	10.685	8.990	18,86%	511	498	2,52%
1973	10.822	9.993	8,29%	526	510	3,09%
1974	10.301	9.767	5,47%	613	594	3,27%
1975	8.805	7.450	18,19%	751	727	3,32%
1976	8.074	7.058	14,39%	672	648	3,70%
1977	8.657	7.507	15,32%	722	695	3,88%
1978	8.129	9.042	-10,10%	758	728	4,17%
1979	6.461	6.023	7,27%	788	760	3,72%
1980	8.260	10.111	-18,30%	652	628	3,79%
1981	8.681	7.379	17,64%	734	712	3,03%
1982	8.698	8.389	3,68%	461	446	3,26%
1983	7.285	8.244	-11,63%	516	500	3,19%
1984	7.419	9.591	-22,65%	525	512	2,61%
1985	6.923	6.817	1,55%	555	545	1,87%
1986	7.408	7.925	-6,52%	449	442	1,66%
1987	7.785	8.051	-3,30%	536	529	1,37%
1988	8.370	9.041	-7,42%	569	563	1,14%
1989	8.195	9.091	-9,86%	476	472	0,78%
1990	7.171	7.220	-0,68%	582	579	0,45%
1991	6.051	6.950	-12,93%	413	412	0,16%
1992	6.252	6.293	-0,65%	319	320	-0,20%
1993	6.278	6.037	3,99%	617	619	-0,26%
1994	6.675	5.798	15,12%	522	523	-0,16%
1995	7.245	5.536	30,86%	510	510	0,01%
1996	6.738	7.691	-12,40%	445	444	0,29%
1997	7.144	6.901	3,52%	998	997	0,10%
1998	6.771	5.768	17,40%	415	415	0,11%
1999	5.941	6.524	-8,93%	474	473	0,26%
Gemiddelde absolute fout			10,96%	Gemiddelde absolute fout		1,87%

(bronstatistiek: CBS, ABF)

## Bijlage 13.4 Modelfouten hindcast arbeidsmarkt Utrecht

Tabel-Bijlage 13.7 geeft inzicht in de fout van het model ten aanzien van het aantal bedrijfsvestigingen in Utrecht.

*Tabel-Bijlage 13.7 Modelfouten hindcast bedrijfsvestigingen Utrecht*

jaar	Bedrijfsvestigingen model	Bedrijfsvestigingen statistiek	Fout bedrijfsvestigingen
1994	45.158	46.240	-2,34%
1995	46.314	45.890	0,92%
1996	47.439	47.550	-0,23%
1997	48.564	49.955	-2,79%
1998	49.688	53.775	-7,60%
1999	50.826	54.300	-6,40%
Gemiddelde absolute fout			3,38%

(bronstatistiek: Statline, CBS)

Tabel-Bijlage 13.8 geeft inzicht in de fout van het model ten aanzien van de potentiële, Utrechtse beroepsbevolking.

*Tabel-Bijlage 13.8 Modelfouten hindcast potentiële beroepsbevolking Utrecht*

jaar	Potentiële beroepsbevolking model	Potentiële beroepsbevolking statistiek	Fout potentiële beroepsbevolking
1972	535.232	534.427	0,15%
1973	541.205	543.725	-0,46%
1974	548.257	554.229	-1,08%
1975	556.608	563.098	-1,15%
1976	567.134	574.073	-1,21%
1977	576.354	582.871	-1,12%
1978	585.064	590.265	-0,88%
1979	594.782	599.223	-0,74%
1980	606.138	610.345	-0,69%
1981	618.808	623.383	-0,73%
1982	628.499	633.410	-0,78%
1983	637.838	642.334	-0,70%
1984	647.656	651.424	-0,58%
1985	658.253	661.659	-0,51%
1986	667.810	671.163	-0,50%
1987	677.258	680.361	-0,46%
1988	686.584	690.172	-0,52%
1989	693.761	697.812	-0,58%
1990	700.684	705.153	-0,63%
1991	707.881	712.113	-0,59%
1992	714.700	718.162	-0,48%
1993	721.211	723.456	-0,31%
1994	727.773	728.706	-0,13%
1995	733.048	732.954	0,01%
1996	737.931	736.845	0,15%
1997	744.547	742.539	0,27%
1998	751.365	748.325	0,41%
1999	758.714	754.568	0,55%
Gemiddelde absolute fout			0,58%

(bronstatistiek: CBS, ABF)