

# Het Modelleren van een Cognitieve Agent

voor Agent-based Computational  
Transaction Cost Economics

door  
Diederik C. Kraaikamp  
(Studentnummer: 0780138)  
maart 2003

Begeleiders:  
Prof. Dr. R.J.J.M. Jorna  
Dr. N.A. Taatgen

Kunstmatige Intelligentie  
(Technische Cognitiewetenschappen)

Rijksuniversiteit Groningen

## Samenvatting

Deze scriptie maakt aanpassingen op een bestaand ACTCE (Agent-based Computational Transaction Cost Economics) model. We onderzoeken of het mogelijk is om de actoren in dit model een cognitief plausibele invulling te geven met behulp van ACT-R. (ACT-R is een architectuur voor cognitie.) Hiervoor halen we uit de literatuur twee belangrijke eigenschappen van *vertrouwen* aan. Vertrouwen is omgaan met onzekerheid en vertrouwen is gebaseerd op persoonlijke ervaringen. De beperkt rationele actoren geven we mede vorm op basis van de concepten satisficing en instance-based learning. Vertrouwen is onderdeel van cognitie, maar het is niet gemakkelijk cognitief te doorgronden of te formaliseren. De agents worden cognitief minimaal ingevuld, zodat ze in staat zijn om relaties met elkaar aan te gaan. Het grote voordeel is, dat men nu ook in de agents kan kijken. In het model wordt gedrag, zoals vertrouwen, in termen van de inhoud en activatie van representaties uitgelegd. Er komen drie experimenten met dit vernieuwde model aan bod. De resultaten hiervan zijn tegengesteld aan een aantal uitgangspunten van TCE (Transaction Cost Economics). In het bijzonder geldt dat hogere differentiatie niet leidt tot extra 'insourcing'. De resultaten laten bovendien zien, dat elke simulatie een starteffect heeft. Het kost tijd om de markt te leren kennen en een goede partner te vinden. De resultaten tonen het belang aan van het maken van een markt. Wanneer dit initiële aanpassingsproces tot rust komt is de kans dat agents optimale relaties hebben gevormd vrijwel nul. Anders dan TCE aanneemt is er veel meer coördinatiecapaciteit nodig om tot optimale resultaten te komen dan de individuele agents bezitten. Hiermee is aangetoond dat de uitgangspunten van TCE niet zo vanzelfsprekend zijn als wordt gedacht. Waar in de toekomst naar gekeken moet worden, is de verdere uitbreiding van het agentmodel, zodat het gedrag van actoren nog plausibeler wordt.

# Inhoudsopgave

<b>Voorwoord</b>	<b>6</b>
<b>1 Inleiding</b>	<b>7</b>
1.1 Rationele analyse en vertrouwen . . . . .	8
1.2 Multi-agent systemen, wat zijn agents? . . . . .	10
1.3 Een agent-based benadering en equilibria . . . . .	11
1.4 Methode van aanpak . . . . .	12
1.5 Opzet van de scriptie . . . . .	13
<b>2 Transaction Cost Economics</b>	<b>14</b>
2.1 Transactiekosten . . . . .	15
2.2 Eigenschappen van actoren . . . . .	15
2.3 Eigenschappen van transacties . . . . .	16
2.4 Complexiteit op actorniveau . . . . .	16
<b>3 Coöperatie, vertrouwen en zekerheid</b>	<b>18</b>
3.1 Sociale dilemma's: een prisoners' dilemma . . . . .	19
3.2 Theory of Mind . . . . .	20
3.3 Definities van vertrouwen . . . . .	22
3.4 Formalisatie van vertrouwen . . . . .	23
<b>4 Een ACTCE model</b>	<b>25</b>
4.1 De Cobb-Douglas vergelijking . . . . .	28
4.2 Het matchingsalgoritme . . . . .	30
4.3 Aanpassingen van het model . . . . .	31

<b>5</b>	<b>Het Agentmodel</b>	<b>33</b>
5.1	ACT-R eisen . . . . .	33
5.2	Satisficing . . . . .	35
5.3	'Vertrouwen' in de simulatie . . . . .	36
5.4	Instance-Based Learning . . . . .	37
5.5	Aanpassingen op het vorige ACTCE-model . . . . .	37
5.6	Een ACT-R agentmodel . . . . .	38
5.7	ACT-R in een ACTCE model . . . . .	41
<b>6</b>	<b>Resultaten</b>	<b>43</b>
6.1	Een TCE experiment . . . . .	43
6.2	Een kijkje in de agent . . . . .	48
6.3	Een tweede experiment, met acceptancequota . . . . .	50
6.4	Derde en laatste experiment, met ruis . . . . .	51
<b>7</b>	<b>Conclusies</b>	<b>57</b>
7.1	Interpretatie van resultaten . . . . .	57
7.2	Initiël aanpassingsproces . . . . .	59
7.3	Verder onderzoek . . . . .	59
	<b>Bibliografie</b>	<b>61</b>
<b>A</b>	<b>Het ACT-R model</b>	<b>67</b>
<b>B</b>	<b>ACT-R, Object Georiënteerd</b>	<b>72</b>
<b>C</b>	<b>ACT-R, de theorie</b>	<b>76</b>
C.1	Goalstack . . . . .	77
C.2	Procedureel geheugen . . . . .	78
C.3	Declaratief geheugen . . . . .	80

# Lijst van figuren

4.1	Relaties op de markt. . . . .	25
4.2	Efficiëntie- en differentiatie-effecten. . . . .	26
5.1	Visions of rationality. . . . .	35
6.1	Insourcing percentage voor verschillende differentiaties. . . . .	45
6.2	Percentage relatievoortzettingen voor verschillende differentiaties. . . . .	46
6.3	De gemiddeld behaalde winst van een koper. . . . .	47
6.4	Genormaliseerd voor een constante relatie met de voordeligste leverancier. . . . .	47
6.5	Genormaliseerd voor voortdurende insourcing. . . . .	48
6.6	Een Type A agent. . . . .	49
6.7	Een Type B agent. . . . .	51
6.8	Proportie insourcing. . . . .	52
6.9	Gemiddeld behaalde winst. . . . .	52
6.10	Chunks met verschillende ruisvarianties. . . . .	53
6.11	Een Type A agent bij veel ruis. . . . .	54
6.12	Een Type B agent bij weinig ruis. . . . .	55
6.13	Insourcing proporties. . . . .	55
6.14	Proportie relatievoortzettingen. . . . .	56
6.15	Gemiddeld behaalde winst. . . . .	56
B.1	Objectmodel met vijf classes hoog in de hiërarchie. . . . .	72
B.2	De ACT-R architectuur als een objectmodel. . . . .	73
B.3	Objectmodel met verschillende chunks. . . . .	74
B.4	Een scenarioview van het beslissingsproces van een actor. . . . .	75

C.1 Een overzicht van de ACT-R architectuur. . . . .	77
C.2 goalslots zorgen voor activatie bij chunks. . . . .	82

# Lijst van tabellen

2.1	Het koppelen van organisatievormen aan transacties. . . . .	17
3.1	Een payoff matrix voor het prisoners' dilemma. . . . .	20
4.1	Een voorbeeld van een set voorkeursvectoren. . . . .	31
5.1	Een geheugenchunk in het model. . . . .	38
5.2	Een productieregel uit het model. . . . .	39
6.1	Parameters voor experiment 1. . . . .	43
6.2	Parameters voor experiment 2. . . . .	50
6.3	Parameters voor experiment 3. . . . .	53
C.1	Een voorbeeld van een goal. . . . .	78
C.2	Een voorbeeld van een productieregel. . . . .	79
C.3	Een voorbeeld van een fail-productie. . . . .	79
C.4	Een voorbeeld van een geheugenchunk. . . . .	81

# Voorwoord

Hoewel dit een veelzijdig (en langdurig) afstuderen is geworden, kan ik oprecht zeggen dat het een plezierige ervaring was. Het is erg fijn om naast het werken binnen een team, toch erg veel vrijheid te hebben om zelf belangrijke beslissingen te maken en daarvan te leren. Daarom wil ik Luit Gazendam, Martin Helmhout en mijn afstudeerbegeleider René Jorna dan ook als eerste bedanken.

Niet alleen bij Bedrijfskunde kon ik terecht voor raad, maar ook bij Kunstmatige Intelligentie, waar Rineke Verbrugge en Niels Taatgen mij een aantal keren op het juiste spoor hebben geholpen. Daarvoor bedank ik beide.

Ook wil ik mijn collega's bij Cosmocows bedanken en in het bijzonder Mathieu van Echtelt, die hoewel hij me waarschuwde voor de omvang van dit afstuderen, interesse toonde en erg behulpzaam was bij het flexibel invullen van mijn werkuren.

Mijn dank gaat ook uit naar mijn vrienden die, hoewel ik ze verveeld moet hebben met uitweidingen over vertrouwen, ACT-R en Transaction Cost Economics, toch koffie met me bleven drinken. En tenslotte dank aan mijn familie.

ACTCE bevindt zich duidelijk op nieuw grondgebied; Het gebied tussen Kunstmatige Intelligentie en Economie, het gebied tussen Psychologie en Informatica, het gebied tussen macro- en micro-wetenschap. Hopelijk is het onderwerp voldoende begrijpelijk gemaakt voor mensen uit al deze gebieden.

Groningen, Maart 2003

Diederik C. Kraaikamp



# Hoofdstuk 1

## Inleiding

”It is the process of becoming  
rather than the never-reached end points  
that we must study if we are to gain insight.”  
(Holland 1992, p. 19)

Deze scriptie is onderdeel van een onderzoeksrichting die Agent-based Computational Transaction Cost Economics (ACTCE) wordt genoemd (Klos, 2000). Het onderzoek is interdisciplinair van aard en integreert kennis vanuit vakgebieden als Economie, Psychologie, Cognitie Wetenschappen en Kunstmatige Intelligentie.

ACTCE ontwikkelt cognitieve modellen waarmee economische theorieën worden getoetst. Eén zo'n model wordt op dit moment ontwikkeld aan de Rijksuniversiteit van Groningen en de Erasmus Universiteit van Rotterdam. Een deel van dat project wordt in deze scriptie beschreven.

ACTCE houdt zich bezig met de vorming van structuren voor het coördineren van transacties; Welk bedrijf verricht welke activiteit en hoe is de relatie tussen bedrijven gedefiniëerd? Onderdeel van deze vraag is, of een bedrijf een relatie met een ander bedrijf aangaat, en zo ja met welk bedrijf? Het bijzondere van ACTCE is, dat zij aandacht schenkt aan de rol die de *beperkte rationaliteit*, het *opportunisme* en het *vertrouwen* van een actor hierbij speelt. Het is deze rol die in deze scriptie onder meer aan de orde komt.

ACTCE combineert Agent-based Computational Economics (ACE) met Transaction Cost Economics (TCE). ACE is de computationele studie van de evoluerende wisselwerking tussen autonome agents samengebracht in een economisch model. ACE is een voorbeeld van het toepassen van het Complexe Adaptieve Systemen (CAS) paradigma op de economie (Klos, 2000). CAS kijkt naar systemen die bestaan uit vele individuele elementen. Deze elementen gaan een wisselwerking met elkaar aan en bezitten tevens de mogelijkheid hun gedrag

hierop aan te passen. De Multi-Agent Systemen (MAS) die we kennen vanuit de Kunstmatige Intelligentie vallen ook binnen deze definitie.

Transaction Cost Economics (TCE) is een economische theorie die voorspellingen kan doen over de vorm waarin transacties worden georganiseerd (Williamson, 1985). Volgens TCE levert een *rationele analyse* de meest efficiënte transactievorm. Dit is de vorm die in de werkelijkheid zal voorkomen. Minder voordelige vormen zullen de concurrentieslag niet overleven.

De genoemde rationele analyse is niets meer dan een analyse van de kosten die komen kijken bij verschillende organisatievormen. Deze kosten hangen af van bepaalde eigenschappen van de transactie in kwestie, zoals de frequentie van het aantal voorkomens, de onzekerheid waarmee hij wordt omgeven en de mate van transactiespecifieke investeringen. De analyse levert een optimale organisatievorm voor transacties voor elke combinatie van eigenschappen. Deze eigenschappen zullen later nog aan bod komen.

ACTCE echter kijkt niet naar een rationele analyse maar naar computersimulaties van, bijvoorbeeld in ons geval, meerdere bedrijven samengebracht in een model. Door te kijken naar de processen van samenwerking en aanpassing, die met behulp van de simulaties naar voren komen, probeert ACTCE dezelfde economische activiteiten op een andere manier te begrijpen. Door uit te gaan van agent-based simulaties kunnen we kijken naar hoe de marktselectie werkt, hoe een markt wordt gevormd, hoe agents leren organisatievormen te kiezen, wat voor (sub)optimale populaties er ontstaan, of we voorspellingen kunnen maken en of we de uitkomsten kunnen beïnvloeden.

De paragrafen verder in dit hoofdstuk gaan dieper in op onderwerpen die hierboven genoemd zijn. Aan het eind van dit eerste hoofdstuk moet duidelijk zijn wat de uitgangspunten van ACTCE en deze scriptie zijn.

## 1.1 Rationele analyse en vertrouwen

De keuze van een organisatievorm, behorende bij een bepaalde transactie, wordt dus gedicteerd door de theorie van TCE. Hieraan ligt een rationele analyse van de situatie aan ten grondslag. Waarom verwerpt ACTCE de rationele analyse terwijl het zo een belangrijk onderdeel is van TCE? TCE omvat zoals gezegd een rationele analyse van organisatievormen en vertelt welke economisch optimaal zijn voor welke transactie. Daarbij gaat TCE uit van twee belangrijke eigenschappen van actoren, ze zijn beperkt rationeel en ze zijn opportunistisch. Beide uitgangspunten hebben echter hun beperkingen. Ten eerste zijn beperkt rationele actoren maar beperkt instaat om zelf een rationele analyse uit te voeren. Ten tweede vertonen actoren vaak ook vertrouwen in en loyaliteit tegenover elkaar en handelen niet altijd opportunistisch. Uitgangspunt binnen ACTCE is daarom dat mensen niet deze rationele analyse aanwenden, maar veeleer een combinatie van simpele heuristieken toepassen (Todd en Gigerenzer,

2000). Heuristieken die door interactie met andere actoren zullen leiden tot al dan niet stabiele vormen van organisatie.

Hoewel een rationele analyse de meest optimale economische organisatievorm uitrekent en TCE beweert, dat deze organisatievorm zal worden toegepast in de werkelijkheid, beweert zij echter niet dat de individuele actoren in staat zijn een dergelijke analyse uit te voeren. Integendeel, TCE zegt juist dat de economische actoren beperkt rationeel en potentieel opportunistisch zijn. Dat deze ogenschijnlijke tegenstelling in TCE geen probleem oplevert, komt door de veronderstelling dat suboptimale organisatievormen door marktmechanismen uitgeselecteerd worden. Actoren die ze gebruiken gaan ten onder in de 'struggle for profit' (Friedman, 1953). Actoren hoeven een rationele analyse dus niet te kunnen uitvoeren om optimale vormen aan te nemen. Tomas Klos (2000, p. 4) maakt duidelijk dat dit argument niet geldt, vanwege verschillende redenen.

"Essentially, the market is not completely efficient as a selection mechanism; an evolutionary selection process does not eliminate all forms of behaviour based on trust and cooperation, like TCE suggests (Williamson, 1985); finally, an evolutionary selection mechanism is not even appropriate in the first place. (...) there are several reasons why biological and economic natural selection can not be compared."

Wat ACTCE duidelijk probeert te maken is dat de werking van de marktselectie niet voldoende is om suboptimale vormen af te doen als niet voorkomend. Een evolutieproces leidt niet tot optimaal gedrag als dit optimale gedrag niet op voorhand in de populatie aanwezig is. Bovendien worden niet alle gedragingen die zijn gebaseerd op vertrouwen en samenwerking geëlimineerd door dit selectieproces. Economische natuurlijke selectie resulteert niet per definitie in de best denkbare oplossing, maar selecteert slechts de beste die in de populatie aanwezig is<sup>1</sup>. Rationele analyse levert echter wel de best denkbare oplossing.

ACTCE wijst redenen aan waarom natuurlijke en economische selectie niet met elkaar te vergelijken zijn. Uit Miller (1996) blijkt dat coöperatie en vertrouwen zich kunnen ontwikkelen binnen een populatie evoluerende actoren. Miller gebruikte een genetisch algoritme om een populatie strategieën voor het spelen van het 'repeated prisoner's dilemma'<sup>2</sup> te laten evolueren. Hieruit bleek dat het de coöperatieve actoren zijn die op de lange termijn overleven en niet diegenen die volgen uit de rationele analyse van TCE (Klos, 1999). Deze resultaten tonen aan dat het natuurlijke selectie argument niet voldoende is om een rationele analyse te verantwoorden. Wanneer dit argument niet werkt, moeten organisatievormen anders tot stand komen dan TCE beweert.

---

<sup>1</sup>In de evolutietheorie speelt naast selectie ook nog het mechanisme van mutatie een belangrijke rol. Het is de combinatie van deze twee, die heeft kunnen leiden tot steeds complexere organismen.

<sup>2</sup>Het 'repeated prisoner's dilemma' komt verderop in deze scriptie uitgebreider aan bod.

Actoren zijn potentiëel opportunistisch. Dit betekent voor TCE, dat een actor in staat moet worden geacht op elk moment zijn eigenbelang na te streven, geheel of gedeeltelijk te liegen, en feiten achter te houden. Samen met de eigenschap van beperkte rationele vermogens levert dit mogelijke situaties op, waarbij de agents geen rekening meer kunnen houden met alle (nadelige) gevolgen van een gebroken relatie.

ACTCE vindt niet dat het aannemelijk is dat, zoals TCE voorspelt, de meest optimale organisatievorm voor een transactie nog zal worden gebruikt, terwijl de individuele actoren zelf geen rekening kunnen houden met alle situaties die zich bij een transactie kunnen voordoen. Daarom introduceren we nog een eigenschap van actoren, een waarvan TCE het bestaan niet ontkent, maar er verder geen rekening mee houdt in de rationele analyse, namelijk *vertrouwen*. Vertrouwen moet de actoren in staat stellen om om te gaan met onzekerheden.

## 1.2 Multi-agent systemen, wat zijn agents?

Volgens ingewijden is een 'agent'<sup>3</sup> een stuk software met eigenschappen die grenzen aan Kunstmatige Intelligentie. In de Kunstmatige Intelligentie komen we een aantal vaakgenoemde eigenschappen van agents tegen. Knapik en Johnson (1998) noemen vier samenvattende eigenschappen:

- Onafhankelijkheid of autonomie: *"a system is autonomous to the extent that its behavior is determined by its own experience."* (Russel en Norvig, 1995, p.35).
- Sociale capaciteiten: agents werken samen of gaan een wisselwerking aan met andere agents of mensen.
- Reactiviteit: agents nemen hun omgeving waar door sensoren en reageren hierop door effectoren.
- Proactiviteit: agents reageren niet slechts reflexmatig maar laten hierbij doelgericht gedrag zien.

Binnen het paradigma van CAS is de agent een metafoor voor een organisme, met als doel het zo waarheidsgetrouw mogelijk modelleren van de te onderzoeken eigenschappen van het organisme, in ons geval is dat organisme de mens.

Veel processen op macroniveau die we terugvinden in sociale systemen, ontstaan door de wisselwerking tussen grote aantallen individuele agents op microniveau.

---

<sup>3</sup>De termen 'agent' en 'actor' lopen in de literatuur door elkaar heen. Carl Hewitt heeft het in 1977 al over 'actors' in de betekenis van computersoftware. Persoonlijk maak ik onderscheid tussen de begrippen actor en agent. Dit betekent dat ik zoveel mogelijk de term 'agent' aanhoud in de context van kunstmatige autonome agents (oftewel software en hardware) en de term 'actor' zal gebruiken in de context van menselijk handelen. Dat dit onderscheid op sommige punten zal vervagen, zie ik als een bewijs van overlap tussen beide begrippen.

Een agent wordt hier gezien als een autonoom systeem dat probeert te voldoen aan een verzameling doelen in een complexe en dynamische omgeving. Agents worden geformaliseerd als actoren die beslissingen nemen op basis van hun doelen, informatie die ze hebben over de omgeving, en hun verwachtingen ten aanzien van de toekomst. De doelen, informatie en verwachtingen van een agent worden beïnvloed door interacties met andere agents. Vaak zijn agents adaptief. Dat betekent dat ze in staat zijn hun beslissingsstrategieën en daarmee hun gedrag te veranderen.

Veel onderzoekers geven agents simpele gedragsregels om te onderzoeken hoe gedrag op macroniveau ontstaat uit deze simpele gedragingen op microniveau (Conte, Hegselmann en Terna, 1997). Een voorbeeld is de 'Tit-for-Tat' regel (Axelrod, 1980a, 1980b), die gebruikt wordt om het ontstaan van coöperatie in grootschalige Prisoners' Dilemmas<sup>4</sup> te onderzoeken.

Dergelijke simulaties met simpele agents worden gebruikt om theorieën te formuleren over het gedrag van sociale systemen. Het is echter zinnig je af te vragen in hoeverre deze theorieën processen in de werkelijkheid naar waarheid beschrijven. Hebben de agents een psychologische of cognitieve onderbouwing? Waaruit kunnen we opmaken dat simpele agents dezelfde regels volgen als biologische sociale systemen? En kunnen we wel complexere agents gaan creëren als bij het bouwen van simpele agents deze vragen niet eerst worden beantwoord?

Deze problematiek wordt deels vermeden door gebruik te maken van ACT-R (Anderson en Lebiere, 1998). ACT-R is een geïntegreerde theorie van cognitie en levert een denkkader waarbinnen modellen kunnen worden gecreëerd en getest. ACT-R levert ons een cognitief fundament waarop we taakspecifieke modellen kunnen bouwen. We moeten echter uitkijken dat het probleem niet wordt verschoven naar de formulering van het in ACT-R te ontwerpen model. Literatuuronderzoek naar vertrouwen, opportunisme en beperkte rationaliteit dient daarom ter onderbouwing van het ACT-R model.

### 1.3 Een agent-based benadering en equilibria

ACE-website<sup>5</sup>:

"A central concern of ACE researchers is to understand the apparently spontaneous formation of global regularities in economic processes, such as the unplanned coordination of trade in decentralized

---

<sup>4</sup>Een prisoners' dilemma is een klassiek probleem uit de Speltheorie en wordt vaak gebruikt om coöperatie te analyseren (Axelrod, 1984). Het behandelt die situaties waarin een fundamenteel conflict bestaat tussen wat een rationele keus is voor individuele leden van een groep en voor de groep als geheel. Het helpt ons te begrijpen hoe dergelijke dilemma's opgelost dienen te worden in het voordeel van het algemeen belang. Verderop in deze scriptie zal dieper worden ingegaan op het prisoners' dilemma.

<sup>5</sup><http://www.econ.iastate.edu/tesfatsi/ace.htm>

market economies that economists associate with Adam Smith's invisible hand. The challenge is to explain how these global regularities arise from the bottom up, through the repeated local interactions of autonomous agents channelled through socio-economic institutions, rather than from fictitious top-down coordination mechanisms such as imposed market clearing constraints or an assumption of single representative agents"

Actoren leren, vertrouwen en gedragen zich niet alsof ze gestuurd worden door middel van een onzichtbare hand. Belangrijk om op te merken is dat hier gesproken wordt van 'globale regulariteiten'. Dit houden niet per se equilibria in. Een Equilibrium is slechts een resultaat, iets wat overblijft als een proces tot rust is gekomen (Schelling, 1978). En processen hoeven niet altijd tot rust te komen. Of zoals Holland (1992, p. 19) zegt *"it is the process of becoming rather than the never-reached end points that we must study if we are to gain inside"*.

Als we invloed willen uitoefenen op een proces om een bepaald doel te bereiken, dan hebben we kennis nodig over de manier waarop een systeem zich ontwikkelt. Deze kennis kan o.a. met computersimulaties opgedaan worden (Epstein en Axtell, 1996).

## 1.4 Methode van aanpak

In deze scriptie wordt een computermodel gebouwd waarin agents gebruik maken van 'satisficing', een beperkt rationele zoekmethode om een keuze te selecteren uit een verzameling van mogelijkheden. Dit levert agents op, die beperkt rationeel zijn en niet zoeken naar een optimale oplossing. Tevens moeten de agents zich kunnen aanpassen en leren welke partners ze het beste kunnen kiezen.

De computersimulatie die we bouwen is een voortzetting van een bestaand model, beschreven in het proefschrift van Tomas Klos (Klos, 2000). Ook dit model is ontwikkeld binnen de onderzoeksgroep ACTCE. Deze scriptie zal vanuit dezelfde benadering als het proefschrift van Tomas Klos te werk gaan. Met als onderscheid dat voor actoreigenschappen als beperkte rationaliteit, opportunisme, en vertrouwen, gezocht zal worden naar een cognitief plausibelere invulling van de agents. Dit betekent dat de agents geen gebruik maken van reinforcement-learning, zoals bij Klos, maar gebruik maken van ACT-R (Anderson en Lebiere, 1998) om ze een adaptief karakter te geven.

De simulatie bevat twee soorten agents, kopers en leveranciers, die transacties met elkaar aangaan. Elke agent, koper of leverancier, representeert één bedrijf. Kopers produceren elke simulatietijdstap een product, waarvan ze de componenten of zelf kunnen fabriceren, of kunnen aanschaffen bij een leverancier. Elke tijdstap moeten kopers daarom kiezen of ze een transactie binnen hun eigen bedrijf organiseren (fabriceren van de componenten, ook wel 'insourcing' genoemd), of een relatie aangaan met een van de vele leveranciers (aanschaffen van de componenten, ook wel 'outsourcing' genoemd). Ook de leveranciers

hebben een keuze, zij kunnen kiezen om al dan niet te handelen met bepaalde kopers. De keuzes die agents moeten maken simuleren we met behulp van het ACT-R model, dat behandeld wordt in hoofdstuk 5.

In tegenstelling tot TCE legt ACTCE met zijn benadering meer nadruk op het proces van organiseren. We houden er rekening mee, dat de optimale vormen die volgens TCE ontstaan niet gevonden zullen worden. Ook hoeft het niet zo te zijn dat het aanpassingsproces convergeert naar stabiele oplossingen. Onafhankelijk van de uitkomsten van de simulatie, wordt hoe dan ook extra inzicht in dit proces vergaard. Niet in de laatste plaats omdat dankzij ACT-R in het brein van de individuele agents gekeken kan worden en van dichtbij kunnen zien hoe ze zich aanpassen en leren.

## 1.5 Opzet van de scriptie

Allereerst wordt in deze scriptie in hoofdstuk 2 een beschrijving van TCE en hoe deze theorie organisatievormen aan transacties koppelt gegeven. Daarna in hoofdstuk 3 wordt het reeds bestaande ACTCE model (Klos, 2000), waarop in deze scriptie wordt voortgebouwd, uit de doeken gedaan. Dit bestaande model bekijken we vooral op het punt van vertrouwen, beperkte rationaliteit, en opportunisme. Hierna kunnen we punten identificeren die we willen veranderen. Deze aanpassingen willen we doen aan de hand van een literatuuronderzoek naar vertrouwen, opportunisme en beperkte rationaliteit, dat wordt beschreven in hoofdstuk 4. Hieruit wordt een theorie over vertrouwen gedestilleerd die we kunnen toepassen. Op deze manier wordt een cognitief en psychologisch geloofwaardig kader voor het model aangeleverd. Met datzelfde kader in gedachten zullen we in hoofdstuk 5 ook de begrippen 'satisficing', een cognitieve theorie over probleemoplossen, en 'instance-based learning', een theorie over leren, introduceren. Aan deze cognitieve onderbouwing wordt verder bijgedragen door te kiezen voor een implementatie van de agents gebaseerd op ACT-R. Uiteindelijk hebben we dan een computersimulatie waarmee we een drietal experimenten zullen uitvoeren. De experimenten en de resultaten bespreken we in hoofdstuk 6. We kijken o.a. naar hoe de agents een markt maken, deze verkennen, en erop handelen. Tenslotte wordt in hoofdstuk 7 de conclusie besproken, en er worden een aantal aanbevelingen voor verder onderzoek gedaan.

## Hoofdstuk 2

# Transaction Cost Economics

Klos (2000) zegt in zijn proefschrift:

”In any case, goods and services have to be transferred from one stage to another. The main substantive theory used here is transaction cost economics, which says what the best structural form is for organizing such transfers. For example, both stages might be brought together within the bounds of a single firm -i.e. vertically integrated-or put in separate firms, using the market to organize transactions between them. A company might own its own R&D department, or outsource research work to a specialized institute. Depending on the characteristics of the transaction, some organizational forms are more appropriate than others. Transaction cost economics is about finding the most appropriate -i.e. the economic-organizational form for a given transaction.”

Transaction Cost Economics zegt welke organisatie het beste bij welke transactie past. TCE is een theorie over de toekenning van organisatiestructuren aan transacties. *”A transaction occurs when a good or service is transferred across a technologically separable interface. One stage of activity terminates and another begins”* (Williamson 1981a, p. 552). TCE richt dus de aandacht op transacties tussen verschillende activiteiten en ziet een bedrijf als één van de mogelijke vormen waarin transacties kunnen verschijnen.

TCE ontstond na Coase' (1937) artikel *The Nature of the Firm*, waarin hij zich afvroeg waarom bedrijven überhaupt ontstaan in een gespecialiseerde exchange economie. Waarom een reeks transacties bij elkaar onderbrengen in één bedrijf, terwijl specialisatie op één specifieke transactie in een gespecialiseerde markt de meest economisch gunstige methode lijkt? Om deze vraag te verduidelijken wordt in de paragrafen hierna dieper ingegaan op transactiekosten, de eigenschappen van transacties, de eigenschappen van actoren, en de complexiteit die dit met zich meebrengt.



## 2.1 Transactiekosten

Een transactie wordt uitgevoerd tussen actoren. Bij een transactie kan men drie fasen onderscheiden: contact, contract en controle (Nooteboom, 1994). Onderhandelingen houden, contracten opstellen, en controlestructuren vormen en uitvoeren, zijn maar al te vaak een bittere noodzaak om een transactie in goede banen te leiden. Al deze handelingen brengen kosten met zich mee, dit zijn de zogenaamde transactiekosten. Deze kosten kunnen vaak verminderd of zelfs geëlimineerd worden wanneer bijvoorbeeld een bedrijfsstructuur wordt gebruikt in plaats van een marktorganisatie. Waar in het laatste geval telkens aparte contracten dienen te worden gevormd, waarin staat wat te doen onder welke omstandigheden, kunnen in het eerste geval beide partijen overeenkomen te gehoorzamen aan een hoger geplaatste manager, die conflictsituaties oplost in het belang van het bedrijf. Als de partijen gescheiden en autonoom zijn, zullen ze echter de voorwaarden in het contract volgens hun eigenbelang willen opstellen.

Een transactie binnen een bedrijf, in plaats van tussen bedrijven, is echter niet altijd het gunstigst. Voor verschillende transacties zijn verschillende organisatievormen het best geschikt om de kosten te minimaliseren. Het zijn de eigenschappen van de actoren en de eigenschappen van de transactie, die de meest geschikte vormen bepalen.

## 2.2 Eigenschappen van actoren

Volgens TCE zijn mensen, de actoren, beperkt rationeel en bovendien opportunistisch. Beperkte rationaliteit wil zeggen dat actoren de intentie hebben om rationeel te handelen, maar slechts beperkt hiertoe de capaciteiten hebben. Ze zijn niet in staat om alle relevante informatie te verzamelen of te verwerken. Opportunisme is het nastreven van eigenbelang door middel van welke methode dan ook, inclusief list en bedrog. Williamson voegt "*calculated efforts to mislead, distort, disguise, obfuscate, or otherwise confuse*" toe, om duidelijk te maken dat de actor actief zijn eigenbelang zal proberen te realiseren (Williamson 1985, p. 47). Er wordt echter niet gesuggereerd dat iedere actor altijd opportunistisch zal handelen, maar slechts dat de een hier meer toe geneigd is dan de ander. Bovendien is het onmogelijk om de actoren hierop van tevoren te scheiden, omdat dit een te complex probleem voor actoren zou zijn. In de rationele analyse van TCE moet er dan ook vanuit worden gegaan dat handelspartners opportunistisch zullen handelen wanneer ze de kans krijgen.

Het is de combinatie van beperkte rationaliteit en opportunisme, die de drie eerder genoemde fasen van een transactie (contact, contract en controle) noodzakelijk maken en gezamenlijk de transactiekosten bepalen. Als de rationaliteit van een actor onbeperkt zou zijn, maakt het niet uit of de partner opportunistisch is, omdat alle mogelijke conflictsituaties beschreven zouden kunnen worden in het contract. En als opportunisme niet zou bestaan, zijn onderhandelingen

en controlestructuren overbodig, omdat alle mogelijke conflictsituaties op een eerlijke manier zouden worden afgehandeld.

## 2.3 Eigenschappen van transacties

Hoe bepaalt TCE nu welke organisatievorm het beste past bij een transactie? Laten we eens kijken naar de eigenschappen van transacties. De dimensies op grond waarvan transacties gekwalificeerd worden binnen TCE zijn, *onzekerheid*, *productspecifieke investeringen*, en *frequentie*.

*Onzekerheid* refereert naar de mogelijkheid, dat zich omstandigheden voor kunnen doen gedurende de transactie, die niet van tevoren konden worden voorspeld en dus ook niet in het contract zijn opgenomen. Op dergelijke momenten kan een actor zich opportunistisch gedragen ten koste van de ander en kunnen er conflictsituaties optreden.

*Productspecifieke investeringen* drukt de beperkte mogelijkheid uit van het inzetten van bepaalde grondstoffen buiten de transactie waarin ze oorspronkelijk geïnvesteerd waren; Autodeuren voor een Ferrari passen niet in een Fiat. Hoe productspecifieker de investeringen zijn des te groter het verlies bij een mislukte transactie. Is er geen sprake van afhankelijkheid vanwege gespecialiseerde goederen, dan kan een transactiepartner vaak zonder al teveel moeite op zoek gaan naar een andere partner in het geval er zich problemen voordoen. Is dit echter wel het geval dan zal er meer geïnvesteerd worden in transactiekosten om een goede afloop te waarborgen.

*Frequentie* heeft te maken met het aantal voorkomens van een transactie. Een transactie die geregeld voorkomt loont zich vaker dan een zeldzame transactie om georganiseerd te worden in de vorm van een bedrijf. Dit is efficiënter omdat het oplossen van eventuele conflictsituaties dan kan gebeuren in het belang van het bedrijf. Bovenstaande eigenschappen bracht Williamson (1979) tot tabel 2.3 (Klos 2000).

## 2.4 Complexiteit op actorniveau

Om de eigenschappen van een transactie te kennen is er informatie nodig. Productspecifieke investeringen en frequentie van de transactie zijn min of meer rechtstreeks af te leiden uit het productieproces. Om de mate van onzekerheid en eventueel opportunistisch gedrag van de partner te kennen is echter informatie van meer persoonlijke aard over die ander nodig.

Het is duidelijk dat individuele actoren allerlei bronnen van informatie gebruiken bij het nemen van beslissingen. Het vereist echter wel dat, wanneer we dit willen onderzoeken, we moeten afdalen naar het niveau van de agents, het microniveau. TCE doet dit niet en zoals is te zien komt de dimensie onzekerheid ook niet in de tabel 2.3 voor.

		Investment Characteristics		
		Nonspecific	Mixed	Idiosyncratic
Frequency	Occasional	Market Governance (Classical Contracting)	Trilateral Governance (Neoclassical Contracting)	
	Recurrent		Bilateral Governance (Relational Contracting)	Unified Governance Contracting

Tabel 2.1: Het koppelen van organisatievormen aan eigenschappen van transacties.

”[i]nasmuch as a great deal of the relevant information about trustworthiness or its absence that is generated during the course of bilateral trading is essentially private information— in that it cannot be fully communicated to and shared with others (Williamson 1975, p. 31-37)—knowledge about behavioral uncertainties is very uneven. The organization of economic activity is even more complicated as a result.” (Williamson 1985, p.59)

Het is deze mate van complexiteit, als gevolg van een gebrek aan informatie over betrouwbaarheid op het actorniveau, waardoor TCE kiest voor een rationele analyse. De mapping van organisatievorm op transactie, kan volgens TCE alleen zichtbaar gemaakt worden door een rationele analyse. Hierbij verliest TCE de kwesies van het maken van, en het zoeken op de markt uit het oog. Het is juist onderzoek op actorniveau, dat noodzakelijk is om deze processen te begrijpen. Binnen deze scriptie wordt daarom wel afgedaald naar het niveau van de individuele actoren. Daarvoor is het nodig te concentreren op een aantal eigenschappen van de actor zoals opportunisme en vertrouwen. Eigenschappen waarvan TCE het bestaan wel erkent, maar ze juist probeert te vermijden.

Er bestaat een eenvoudige relatie tussen opportunisme, vertrouwen en onzekerheid. Onzekerheid aangaande situaties in de toekomst is over het algemeen ongewild maar onvermijdelijk. Helemaal wanneer een actor niet te vertrouwen blijkt te zijn, of opportunistisch is. Wanneer een partner wel betrouwbaar wordt geacht, heft dit de negatieve uitkomsten van een onzekere toekomst op. Met andere woorden, vertrouwen in de ander maakt het mogelijk meer risico te nemen, bijvoorbeeld door een relatie aan te gaan in situaties waar onzekerheid heerst. In het volgende hoofdstuk gaan we dieper in op de rol van deze eigenschappen.

## Hoofdstuk 3

# Coöperatie, vertrouwen en zekerheid

Hoewel sociaal gedrag in de vorm van coöperatie en vertrouwen geen duidelijke voordelen hoeft te hebben, nemen velen aan dat deze vormen van altruïstisch gedrag plaatsvinden omdat het beloond wordt. Over het algemeen geeft een dergelijke daad een goed gevoel. Eén verklaring hiervoor is dat altruïsme is ingebouwd in de mens (Axelrod en Hamilton, 1981).

Ook zonder deze biologische basis kan beredeneerd worden dat sociaal gedrag vaak tot beloning leidt en dus net zo makkelijk aangeleerd als aangeboren kan zijn (McGovern, Ditzian, en Taylor, 1975). Deze reinforcement theorie van pro-sociaal gedrag wordt meer complex wanneer we rekening houden met de cognitieve context. Mensen reageren niet alleen op simpele externe genoegens en ongenoegens, maar ook op overtuigingen, normen en waarden, en verwachtingen over de consequenties van het gedrag, kortom op representaties (Darley en Batson, 1973).

Een belangrijke eigenschap die invloed heeft op pro-sociaal gedrag is de neiging tot empathie, dat wil zeggen de geneigdheid om te handelen vanuit het perspectief van anderen (Eisenberg en Miller, 1987). Wanneer men simpelweg bewust is van een ander zijn problemen, kan men sympathie voelen. Wanneer men probeert die ander zijn subjectieve ervaring te begrijpen, is er sprake van empathie (Wispe, 1986). In het algemeen als we iemand zien lijden, voelen we persoonlijke stress en bezorgdheid of empathie voor het slachtoffer. Echter, alleen empathie leidt tot altruïstisch gedrag (Batson, Fultz, en Schoenrade, 1987; Batson e.a., 1983). Diegenen met veel empathie zijn behulpzaam zelfs als niemand, inclusief het slachtoffer, weet van hun actie. De motivatie lijkt duidelijk intern te zijn. Desalniettemin kan betoogd worden dat zelfs het meest altruïstische gedrag gebaseerd is op zelfzuchtige motieven (Cialdine e.a., 1987).

TCE gebruikt de evolutietheorie om te beweren dat alleen optimaal gedrag overblijft dat volgt uit een rationele analyse. Het is deze rationele analyse die

in TCE geen ruimte laat voor vertrouwen. Dit betekent echter niet dat vertrouwen uit de populatie verdwijnt. Integendeel, vertrouwen is vaak nuttig en kan potentieel juist tot optimaal gedrag leiden. De evolutietheorie leert ons dat het belang van het doorgeven van de genen boven alles gaat. Dit belang hangt nauw samen met het belang van het individu, maar we mogen niet vergeten dat dit individuele belang soms grotendeels van een groep kan afhangen. Vertrouwen kan een nuttige rol spelen in elke groep, waarin samenwerken in sommige gevallen economischer is dan niet samenwerken en tot hogere overlevingskansen kan leiden.

### 3.1 Sociale dilemma's: een prisoners' dilemma

Vertrouwen speelt een grote rol bij wat *sociale dilemma's* worden genoemd. Dit zijn situaties waarin een bepaalde actie (1) voordeel oplevert voor de actor; (2) schade berokkent aan diegenen die hem niet uitvoeren; en (3) meer schade veroorzaakt bij iedereen als iedereen hem uitvoert. De ecooloog Hardin (1968) toonde de relevantie aan van sociale dilemma's in het beroemd geworden artikel "*The Tragedy of the Commons*." Hierin vertelt hij het verhaal van een gezamenlijke weide, waarin Engelse dorpingen hun vee konden laten grazen. Terwijl het aantal dieren langzaam de capaciteit van de weide naderde, kwamen de dorpingen voor een dilemma te staan. Een dorping kon een extra koe kopen, wat hem extra inkomsten opleverde. De schade toegebracht aan de weide zou hierbij vergeleken klein zijn, maar als iedere dorping koeien zou toevoegen, zou de weide geruïneerd raken. De tragedie is, dat om de weide te beschermen elke dorping een deel van zijn vrijheid moest inleveren.

Om dit soort situaties te bestuderen heeft men spellen bedacht. De bekendste zijn de zgn. *prisoners' dilemma* spellen. Een prisoners' dilemma kent een aantal regels. Het wordt gespeeld door twee spelers. Een speler kan kiezen tussen twee acties, een coöperatieve of een concurrerende strategie. Geen van beide kent de keuze van de ander totdat beiden hebben gekozen. De beloning voor een speler hangt af van de combinatie van de twee strategieën. Dit wordt handig weergegeven in een payoff matrix, zie tabel 3.1. Deze matrix moet voldoen aan vier eisen: (1) de hoogste gezamenlijke totale beloning vindt plaats wanneer beiden samenwerken, (2) de laagste gezamenlijke totale beloning wanneer beiden concurreren. (3) De hoogste individuele beloning is voor de speler die concurreert terwijl de ander samenwerkt, en (4) de laagste individuele beloning gaat naar diegene die samenwerkt terwijl de ander concurreert.

Wat is nu de beste strategie in dit spel? Om deze vraag te beantwoorden heeft Robert Axelrod (1984) een virtueel toernooi georganiseerd, gespeeld door computerprogramma's, waarbij alle spelers meerdere malen tegen elkaar uitkwamen. De programma's werden ingediend door sociale wetenschappers, wiskundigen, computer whizzkids, en anderen met kennis van spelstrategieën. Het doel was zoveel mogelijk punten verzamelen gedurende het toernooi. Sommige van deze programma's waren in staat alle voorgaande acties van tegenspelers mee te

		Speler 1	
		samenwerken	concurreren
Speler 2	samenwerken	2 \ 2	0 \ 3
	concurreren	3 \ 0	1 \ 1

Tabel 3.1: Een payoff matrix voor het prisoners' dilemma. Wanneer Speler 1 kiest voor samenwerking terwijl Speler 2 kiest voor concurreren, dan krijgt de eerste een beloning van 0 en de tweede een beloning van 3.

nemen in hun overweging. Echter de uiteindelijke winnaar was het simpelste programma.

Dat programma werd ingezonden door psycholoog Anatol Rapoport en heette TIT-FOR-TAT (TFT). Het bestond uit slechts twee regels: (1) de eerste keer dat je een nieuwe speler tegenkomt, werk dan samen. (2) Anders kies je die strategie die de ander de vorige keer speelde. Wat opvalt, is dat TFT niet in staat is te winnen van een ander in een één tegen één situatie. Het werkt nooit minder samen dan de ander. TFT won doordat hij anderen zover kreeg met hem samen te werken. Iets waar de rest onderling minder goed in was.

Volgens Axelrod's analyse zijn er vier oorzaken, waarom TFT zo goed is in het initialiseren van samenwerking: (1) TFT is aardig. Door vanaf het begin samen te werken wordt de ander aangemoedigd hetzelfde te doen. (2) TFT is niet exploiteerbaar. Elke concurrerende actie wordt met gelijke munt terugbetaald. (3) TFT is vergevingsgezind. Het hervat de samenwerking zodra de ander dit ook weer doet. (4) TFT is doorzichtig. Het programma is zo simpel, dat een ander zijn strategie snel doorheeft en tot de conclusie komt dat samenwerking het beste is.

De effectiviteit van deze strategie heeft evolutionaire theoretici aanleiding gegeven om te betogen dat wij een geëvolueerde neiging hebben tot wederkerig gedrag. We zijn van nature geneigd diegenen te helpen die ons helpen en diegenen die ons tegenwerken niet te helpen. We kijken uit voor bedriegers en straffen of vermijden hen als we kunnen. Toch vergeven we vroegere bedriegers die berouw hebben en we gaan opnieuw met ze samenwerken.

## 3.2 Theory of Mind

Uit onderzoek van Schindler en Thomas (1993), blijkt dat integriteit<sup>1</sup> en competentie de voornaamste eigenschappen zijn waarnaar een individu zoekt bij het bepalen van een ander zijn betrouwbaarheid. Integriteit lijkt het hoogst gewaardeerd te worden, omdat zonder een gewaarwording van de ander zijn morele karakter (normen en waarden) en eerlijkheid, de overige eigenschappen van een individu betekenisloos zijn. De grote importantie van competentie is

<sup>1</sup>Integriteit definiëren Schindler en Thomas als eerlijkheid en waarheidsgetrouw zijn.

waarschijnlijk het gevolg van de behoefte van teamleden naar samenwerking met gelijken om hun verantwoordelijkheden succesvol uit te voeren. Iemand vertrouwen die weinig of niets voor je kan betekenen is veelal nutteloos.

Deze eigenschappen, integriteit en competentie, zien we weerspiegeld in een veelgemaakte opdeling van vertrouwen. Vertrouwen in de *competentie* van iemand in bepaalde omstandigheden. En vertrouwen in de *intentie* van een persoon (Nooteboom, 1999b). Deze opdeling is ook te maken in het begrip onzekerheid. Aan de ene kant bestaat er onzekerheid over de mogelijkheden of situatie en aan de andere kant weten we niet wat iemand zijn intenties of motieven werkelijk zijn.

Samenwerken brengt dus onzekerheid met zich mee over de intenties en competenties van een ander. De mate van onzekerheid varieert. Men kan compleet vertrouwen hebben in het toekomstige gedrag van iemand, waarbij onzekerheid geen rol speelt. En men kan compleet in het duister tasten. Meestal echter is er sprake van een tussenvorm, waarbij onzekerheid bestaat over de afloop. Vertrouwen is een manier van omgaan met deze onzekerheid. Vertrouwen zorgt ervoor dat we samenwerken en ons afhankelijk opstellen in relatie tot de ander, waar dit zonder vertrouwen niet kan. Wat vertrouwen zo bijzonder maakt is haar ongrijpbare, irrationale, holistische karakter. Dit in tegenstelling tot andere bronnen van zekerheid, waarvan we weten waar ze vandaan komen<sup>2</sup>.

Zoals hierboven is genoemd, wordt vaak onderscheid gemaakt tussen vertrouwen in competentie en vertrouwen in intentie. Simpel gezegd, maakt het een actor niet uit dat de partner slechte intenties heeft, als die toch niet de competentie heeft om naar zijn intenties te handelen. Competentie zal in het model geen rol spelen omdat ze wordt verondersteld altijd aanwezig te zijn. We gaan er namelijk vanuit dat agents de competentie bezitten om relaties aan te gaan, of te beëindigen, met willekeurig welke andere agent en te voldoen aan in- of verkoop. Ons onderzoek richt zich dus op intentioneel vertrouwen. De poging van de agents om een accurate representatie van anderen te ontwikkelen, kan dan ook gezien worden als het maken van een representatie van de intenties van die anderen.

De intentie van een persoon is nooit met zekerheid te meten, toch weerhoudt dit ons niet om hiervan een beeld te vormen bij anderen. Dit komt overeen met wat Castelfranchi en Falcone (1999) zeggen. Zij leveren een cognitieve kijk op vertrouwen als een complexe structuur van overtuigingen en doelen implicerend dat er sprake is van een 'theory of mind' van de partner. Ze houden rekening met de mentale attitudes zoals een geloof in een actors mogelijkheden, kansen, vermogens, en bereidheid.

Vertrouwen bestaat uit overtuigingen, evaluaties, en verwachtingen over de ander zijn mogelijkheden, zelfvertrouwen, bereidheid, stabiliteit, morele waar-

---

<sup>2</sup>Een model van cognitie voorgesteld door Epstein c.s. (Denes-Raj en Epstein, 1994) biedt hiervoor een verklaring. Volgens dit model, hetende Cognitive-Experiential Self-Theory (CEST), proberen wij de wereld op twee manieren te begrijpen. De ene methode is bewust, rationeel denkend, en volgt regels van de logica. De andere is meer intuïtief en werkt op een automatische, holistische wijze.

den, motivaties, doelen, en overtuigingen. Aan overtuigingen, evaluaties, en verwachtingen gaan niet altijd bewuste beslissingsprocessen vooraf. Voor de vertrouwende persoon komen ze over als gegeven feiten. Vertrouwen kent deze eigenschap ook, in ieder geval voor een deel<sup>3</sup>. Andere bronnen van informatie waarop vertrouwen kan zijn gebaseerd, zijn persoonlijke neigingen zoals goedgelovigheid of achterdocht.

Op basis van welke informatie vormen wij een theory of mind van een ander? Mensen gebruiken diverse bronnen voor de verzameling denkbeelden waarop vertrouwen gebaseerd is. Belangrijke bronnen kunnen zijn persoonlijke ervaringen en persoonlijke interacties. Daarbij komen nog observaties, redeneringen, sociale stereotypen, communicatie, spreiding van reputatie. Al met al een hele verzameling.

### 3.3 Definities van vertrouwen

Vertrouwen is een onderdeel van de cognitie waarvan we niet in staat zijn om het compleet bewust en rationeel te maken, maar het is net zo goed onderhevig aan dezelfde eisen die normaal gesproken aan cognitie worden gesteld. Gedacht moet worden aan eisen die bijvoorbeeld worden gedictieerd door beperkte rationaliteit, beperkt geheugen, beperkte informatie, of beperkte hoeveelheid beslissingstijd.

Omdat vertrouwen zo'n veelzijdig begrip is, is zij moeilijk te doorgronden. Williamson (1985) vindt dat we het vage menselijke concept vertrouwen moeten elimineren, als het slechts gaat over subjectieve kansen bij het maken van beslissingen. Vertrouwen is een manier om om te gaan met onzekerheid. Hij zegt, dat door te vertrouwen we geloven in een bepaalde afloop van de toekomst. Vertrouwen is volgens hem computationeel, dat wil zeggen een cognitieve handeling die kwantitatief te modelleren is. Dus het concept vertrouwen kan geformaliseerd worden door subjectieve waarschijnlijkheden toe te kennen aan toekomstige gebeurtenissen<sup>4</sup>

Deutsch (1958) formuleert vertrouwen als volgt. Een individu heeft vertrouwen in het plaatsvinden van een bepaalde gebeurtenis, als hij deze verwacht en als zijn verwachtingen leiden tot gedrag waarvan hij denkt, dat dit negatievere consequenties zal hebben in het geval zijn verwachtingen niet uitkomen dan wanneer ze wel uitkomen.<sup>5</sup> Vertrouwen in een ander kan worden gezien als de subjectieve waarschijnlijkheid dat die ander niet opportunistisch zal handelen (Deutsch 1973, Gambetta 1988).

<sup>3</sup>Hier zien we het holistische karakter van vertrouwen weer.

<sup>4</sup>Kwantitatieve maten voor vertrouwen kunnen ook gevonden worden in Marsh (1994), Axelrod (1984), Boon en Holmes (1991), Birk (1999).

<sup>5</sup>Deutsch: "An individual may be said to have trust in the occurrence of an event if he expects its occurrence and his expectations lead to behaviour which he perceives to have greater negative consequences if the expectation is not confirmed than positive motivational experiences if it is confirmed."



Gulati (1995) zegt echter dat vertrouwen toeneemt met de duur van de relatie. Hoe langer een relatie duurt hoe meer we het gedrag van de partner voor normaal houden. Er ontstaat een band die de relatie in stand houdt.

Het opstellen van een gedetailleerd contract dat weinig ruimte laat voor opportunistisch gedrag, zal niet (zo snel) bijdragen tot een verandering in het vertrouwen in de intenties van een partner. Dit kan verklaard worden door te bedenken, dat een gedetailleerd contract geen ruimte overlaat voor het opdoen van ervaringen, die de actor iets kunnen zeggen over de intenties van de partner. Representaties hierover worden niet opgebouwd. Vertrouwen kan op die manier niet groeien.

Het zijn de ideeën van Numan die goed aansluiten bij wat er tot nu toe behandeld is. Hij zegt dat vertrouwen een manier is om het gevoel van complexiteit, onzekerheid en risico te verminderen (Numan 1998, p. 30). Vertrouwen definiëert hij als volgt:

”Trust is a mental action. This action is an expectation which person A has of an actor B - that actor B will act positively toward the goals which the trusting person A has. In this, the actor B, who has to be trusted, has the freedom to harm the trusting person A. The expectation is based on incomplete evidence.” (Numan, 1998, p. 39)

### 3.4 Formalisatie van vertrouwen

Vertrouwen lijdt onder wat we de informatieparadox kunnen noemen. Vertrouwen is gebaseerd op informatie, maar is slechts relevant als er onzekerheid bestaat, dat wil zeggen als de informatie incompleet is. Als een actor denkt te bezitten over volledige informatie oftewel uitsluitend bewijs, dan spreken we van 'confidence' (zekerheid). Gelooft de actor in een bepaalde loop der gebeurtenissen, terwijl hij niet denkt te beschikken over enig bewijs hiervoor, dan praten we over 'faith' (geloof). Heeft de actor daarentegen het idee over gedeeltelijk bewijs te bezitten, dan hebben we het over 'trust' (vertrouwen).

In de literatuur komen veel verschillende definities van vertrouwen voor, maar een aantal eigenschappen komen vaker naar voren dan andere. Vertrouwen speelt een rol wanneer een keuze gemaakt moet worden die zowel positief als negatief gewaardeerde gevolgen kan hebben. Er is sprake van onzekerheid over de afloop, een zeker risico bij de beslissing. Die onzekerheid zorgt ervoor dat het maken van een keuze bemoeilijkt wordt. Vertrouwen wordt veelal zo geformuleerd dat het de onzekerheid gedeeltelijk of geheel opheft, zodat beslissingen kunnen worden genomen op basis van vertrouwen in plaats van onzekerheid.

Onzekerheid kent verschillende vormen, zo ook vertrouwen. Er kan gesproken worden over vertrouwen in de situatie, vertrouwen in een persoon, of vertrouwen als een karaktereigenschap. De laatste vorm van vertrouwen is wellicht het

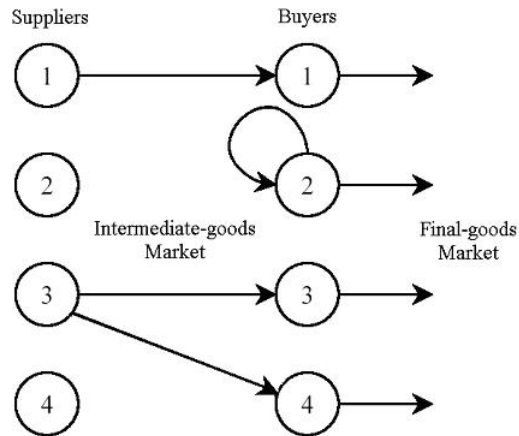
gemakkelijkst te implementeren, namelijk als een globale relatief constante variabele, een zogenaamde base-level voor vertrouwen. Volgens Hill (1990) is dit uit te leggen als een inschatting van de proportie van de bevolking welke nooit opportunistisch gedrag vertoont.

Eén van de bronnen van vertrouwen die we noemden zijn persoonlijke ervaringen. Deze ervaringen worden door de actor opgeslagen als representaties van interacties met anderen. Dit impliceert dat een actor nieuwe representaties creëert, terwijl hij opereert in zijn omgeving. Het is op basis van de verzameling ervaringen die zo ontstaat, dat we denken dat een actor zijn toekomstig handelen kiest. In het nu volgende hoofdstuk kijken we hoe het ACTCE model eruit ziet, dat dit alles moet gaan bevatten. In het hoofdstuk daarna zullen we het geheel implementeren in de vorm van een agentmodel.

## Hoofdstuk 4

# Een ACTCE model

Het doel van deze scriptie is een reeds bestaand model aan te passen op zo'n manier dat het een cognitief meer geloofwaardige invulling krijgt van de agents in het model. Deze invulling proberen we enerzijds te realiseren aan de hand van het in het vorige hoofdstuk behandelde begrip vertrouwen en anderzijds met behulp van de cognitietheorie van ACT-R, die we later in deze scriptie nog zullen tegenkomen. Omdat we voortbouwen op het proefschrift van Tomas Klos (2000) en de afstudeerscriptie van Martin Helmhout (2001), zullen we in dit hoofdstuk eerst de simulatie behandelen die zij (mede) hebben gecreëerd. Aanpassingen op dat model komen daarna aan bod.



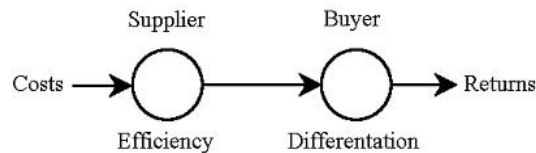
Figuur 4.1: Relaties op de markt. Kopers worden aan zichzelf of aan leveranciers gekoppeld.

De simulatie van Klos richt zich op de intermediate-goods markt, zie figuur 4.1.

Hierin bevinden zich twee soorten agents, leveranciers en kopers. Leveranciers leveren componenten aan de kopers, die een eindproduct produceren dat ze verkopen op de final-goods markt. Kopers mogen kiezen tussen de componenten aanschaffen bij een leverancier (outsourcing) of hun eigen componenten produceren (insourcing).

Dit besluit van de kopers om uit te besteden aan de leveranciers of om zelf te produceren, zal binnen de context van vertrouwen onderzocht worden; Gaat een koper een relatie aan met een ander of wordt de transactie binnen het eigen bedrijf gestalte gegeven? Zijn kopers en leveranciers loyaal aan elkaar, of vertonen ze opportunistisch gedrag door steeds van partner te veranderen?

We hebben een leverancier die componenten verkoopt. En een koper die winst maakt op het eindproduct, een winst die ze samen delen. Voor elke eenheid product die de koper produceert is ook één eenheid component nodig. De kosten van de componenten bedragen één eenheid en de productprijs is ook één eenheid. Dit zou normaal gesproken betekenen dat er geen winst gemaakt wordt. Winst is hier namelijk productprijs minus grondstofkosten. Echter dankzij efficiëntie- en differentiatie-effecten kunnen de agents toch een winst behalen. Zie ook figuur 4.2.



Figuur 4.2: Efficiëntie- en differentiatie-effecten. Alleen de leverancier kent efficiëntie-effecten.

Differentiatie wil zoveel zeggen, dat producten enigszins kunnen afwijken van het standaardproduct zodat de concurrentiepositie wordt verbeterd. Door beter op de vraag in te spelen met een aangepast product kan de waarde worden verhoogd. Gedifferentieerde producten leveren dus meer winst op dan gestandaardiseerde. De sterkte van marktpositie ten gevolge van differentiatie wordt uitgedrukt in de variabele  $d$  waarvoor geldt ( $0 \leq d \leq 1$ ). De formule voor de gemaakte omzet ( $r$ ) op één eenheid product wordt,

$$r_i = 1 + d_i \tag{4.1}$$

Aan de kant van de koper wordt de winst op een eenheid verhoogd door differentiatie van zijn product. De differentiatie is een constante vanuit het gezichtspunt van de agents. Een agent kan in het model van Klos niet zelf een differentiatie van zijn product kiezen.

Een leverancier draagt bij aan de winst door efficiënter de componenten te produceren en beschikt over efficiëntie-effecten die kopers niet hebben.

Een component bestaat voor een deel uit productspecifieke en voor het overige deel uit algemeen inzetbare grondstoffen. We maken dus onderscheid tussen twee typen componenten. Componenten die specifiek zijn voor een bepaalde koper kunnen niet geleverd worden aan andere kopers. En voor componenten die algemeen inzetbaar zijn, maakt het niet uit welke koper ze gebruikt. Aangenomen wordt, dat er een relatie bestaat tussen de hoeveelheid product-specifieke grondstoffen en de mate van differentiatie van het product.

$$k_{ij} = d_i \quad (4.2)$$

Net zoals er twee soorten componenten zijn, zijn er twee vormen van efficiëntie-effecten in ons model, schaalvoordelen en ervaringseffecten.

Het product van de koper bestaat voor  $(1 - d)$  deel uit algemeen inzetbare grondstoffen. Het voordeel van algemeen inzetbare componenten is, dat ze inwisselbaar zijn met de algemeen inzetbare componenten nodig voor producten van andere kopers. Dit maakt ze gevoelig voor schaaffecten. Dat betekent, dat hoe meer van dit type wordt geleverd hoe minder de kosten per eenheid zijn. Schaaffecten treden op wanneer een leverancier zijn productie weet op te voeren door aan meerdere kopers te leveren. Dit levert een totaal van algemeen inzetbare grondstoffen ( $g$ ) gemaakt door leverancier  $j$ .

$$g_j = \sum_{kopers} (1 - d_i) m_{ij} \quad (4.3)$$

$m$  is hierbij 1 als de leverancier aan de koper heeft geleverd en 0 als dit niet het geval is.

Voor  $d$  deel bestaat het product uit koper specifieke grondstoffen. Voor productspecifieke componenten geldt dat ze onderhevig zijn aan ervaringseffecten. Het idee hierachter is, dat het leereffect optreedt omdat de concurrentie het productieproces nog niet heeft geoptimaliseerd<sup>1</sup>. De leereffecten zijn afhankelijk van de tijd, hoe langer een relatie duurt hoe efficiënter een leverancier wordt in het produceren van de productspecifieke componenten. Leereffecten belonen loyaliteit aan een partner. Beide effecten, schaalvoordelen en ervaringseffecten, worden gemodelleerd door dezelfde functie.

$$Y = \max[0, 1 - \frac{1}{(fx + 1 - f)}] \quad (4.4)$$

Voor schaaffecten krijgt deze functie<sup>2</sup> de volgende invulling,

$$e_{s,j} = \max[0, 1 - \frac{1}{(f_s g_j + 1 - f_s)}] \quad (4.5)$$

---

<sup>1</sup>Algemeen inzetbare grondstoffen kennen geen leereffecten omdat er vanuit wordt gegaan dat hun productieproces vanwege hun standaardisatie al zover geperfectioneerd is dat er geen leereffecten meer zijn te behalen.

<sup>2</sup> $f_s$  Is hierin een constante die Klos de scalefactor noemt en krijgt de waarde 0.5.

En voor leereffecten<sup>3</sup>,

$$e_{l,ij} = \max\left[0, 1 - \frac{1}{(f_l x_{ij} \text{dstappen} + 1 - f_l)}\right] \quad (4.6)$$

Leverancier  $j$ 's kosten ( $c$ ) voor het produceren het component nodig door koper  $i$  zijn,

$$c_{ij} = k_i(1 - e_{l,ij}) + (1 - k_i)(1 - e_{s,j}) \quad (4.7)$$

Alles bij elkaar levert het bovenstaande de volgende formule op voor de uiteindelijk behaalde winst ( $p$ ), eventueel verdeeld tussen koper en leverancier.

$$p_{ji} + p_{ij} = (1 + d_i) - (d_i(1 - e_{l,ij}) + (1 - d_i)(1 - e_{s,j})) \quad (4.8)$$

Het hoogste leereffect wordt bereikt als een relatie tussen koper en leverancier oneindig lang duurt. Het hoogste schaaffect bereikt een leverancier wanneer hij zijn volle quotum aan algemeen inzetbare componenten levert. In de simulatie is dit het geval wanneer de leverancier een relatie heeft met zoveel mogelijk kopers. De maximaal te behalen winst wordt dus verkregen wanneer een koper in zee gaat met een leverancier met wie hij een oneindig durende relatie aangaat en die levert aan het hoogste mogelijke aantal andere kopers<sup>4</sup>.

## 4.1 De Cobb-Douglas vergelijking

Elke agent  $i$  kent een score toe aan alle agents  $j$  waar hij mogelijk aan gekoppeld kan worden. Klos kiest voor een zogenaamde Cobb-Douglas vergelijking, die in afweging tussen vertrouwen en winstgevendheid mogelijk maakt.

De winst is van tevoren niet volledig te voorspellen. Winst hangt zoals we zagen niet alleen af van differentiatie. Ze hangt ook af van de twee efficiëntie-effecten, ervarings- of leerefficiëntie en schaafficiëntie. Hoewel de eerste wel te voorspellen is, wetende dat een relatie wordt voortgezet of verbroken, is de tweede pas na het matchingsproces bekend. Dan pas is bekend aan hoeveel kopers een leverancier gekoppeld is en hoe groot de schaalvoordelen zijn.

Om de agents toch een afweging te laten maken tussen opportunisme en vertrouwen kennen ze wel de zogenaamde potentiële winst ( $p$ ) van een bepaalde transactie. Deze maakt een schatting van de te verwachten schaaffecten. De potentiële winst is verwerkt in de formule die de verschillende voorkeurscores berekent.

$$s_{ij} = (p_{ij})^{\alpha_i} \cdot (t_{ij})^{1-\alpha_i} + \tau \quad (4.9)$$

<sup>3</sup>  $f_l$  Is hierin een constante die de learnfactor wordt genoemd. Deze wordt op ook 0.5 gezet.

<sup>4</sup> Het is deze maat die door Klos wordt gebruikt om te kijken of optimale uitkomsten worden gevormd. Er kunnen hier echter kanttekeningen bij worden geplaatst. In veel simulaties levert insourcing de meeste winst op, leer- en schaaffecten wegen dan namelijk niet op tegen het afstaan van de helft van de winst.

Deze formule heeft nog drie andere belangrijke variabelen. Vertrouwen  $t$  groeit met de tijd en heeft bovendien een geheugeneffect.<sup>5</sup>

$$t_{ij} = t_{init,ij} + (1 - t_{init,ij}) \left( 1 - \frac{1}{f_t x_{tijdstappen} + 1 - f_t} \right) \quad (4.10)$$

De parameter  $\alpha$  in vergelijking 4.9 zegt iets over het belang dat een agent hecht aan vertrouwen versus opportunisme, is  $\alpha$  hoog dan speelt vertrouwen een kleine rol en is  $\alpha$  laag dan heeft de verwachte winst weinig invloed.  $\tau$  Zegt iets over loyaliteit aan de huidige partner, deze waarde wordt dan ook alleen bij hem opgeteld.  $\tau$  Zorgt ervoor dat de huidige leverancier kan concurreren met de keuze van de koper om voor zichzelf te gaan produceren. Zonder  $\tau$  zou de voorkeurscore te gemakkelijk in het voordeel van insourcing uitvallen, vanwege de winst die dan niet gedeeld hoeft te worden.

Het adaptieve karakter van de agents in het model van Klos verwijst naar de mogelijkheid om elke tijdsstap de waarden van  $\alpha$  en  $\tau$  te kiezen. De agent kan kiezen uit een aantal discrete waarden die binnen een interval liggen;  $\alpha \in \{0, 0.2, 0.4, 0.6, 0.8, 1\}$  en  $\tau \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5\}$  bijvoorbeeld<sup>6</sup>. De verschillende mogelijke waarden voor  $\alpha$  en  $\tau$  kennen elk een gewicht, de gewichten sommeren tot twee constante waarden,  $C_\alpha$  en  $C_\tau$ . De kans dat aan het begin van elke tijdsstap bepaalde waarden voor  $\alpha$  en  $\tau$  worden gekozen, is proportioneel aan het gewicht dat toegekend wordt aan deze waarden. Door nu elke tijdsstap de gewichten met behulp van reinforcement-learning aan te passen, zouden waarden die tot hogere opbrengsten leiden vaker uitgekozen worden. Om dit te bereiken wordt een constant deel van de gemaakte winst aan het eind van elke ronde opgeteld bij het gewicht dat hoort bij de aan het begin gekozen waarde voor  $\alpha$  en  $\tau$ . En daarna de gewichten weer te normaliseren, zodat ze samen weer  $C_\alpha$  en  $C_\tau$  tellen.

Stel je een rad van fortuin voor met net zoveel vakken als er waardes zijn. De grootte van elk vak is proportioneel aan het gewicht van de waardes, de sterkste waarde heeft het grootste vak. Het rad wordt elke tijdstap rondgedraaid en zo worden de te gebruiken waarden voor  $\alpha$  en  $\tau$  bepaald. Het gewicht van de gekozen waarde wordt telkens aangepast op basis van de gehaalde winst.

De grootste gewichten zullen steeds meer het gewogen gemiddelde hun richting op trekken. Een bepaalde initialisatie kan dus erg veel invloed uitoefenen op de scores en dus de voorkeursvector. Dit verschijnsel zou geen grote gevolgen hebben als we te maken zouden hebben met een min of meer statisch fitness-landschap. Ons landschap is echter niet statisch omdat ze gekoppeld is aan de keuzes van andere agents. Hierdoor kunnen initialisaties veel invloed uitoefenen op de resultaten.

Het nadeel van de algemene aard van  $\alpha$  en  $\tau$  is, dat het niet mogelijk is om het vertrouwen wat alle andere agents in één bepaalde agent hebben aan te passen.

<sup>5</sup> $f_t$  Is in deze formule een constante. En  $t_{init,ij}$  is het initiële vertrouwen van agent  $i$  in  $j$ .

<sup>6</sup>Klos gebruikt voor zijn experimenten vijf in plaats van zes waarden binnen het interval  $[0, 1]$  voor  $\alpha$  en  $[0, 0.5]$  voor  $\tau$ .

Zoals te zien in de Cobb-Douglas formule kent  $\alpha$  alleen de index  $i^7$ . Dit betekent dat er geen onderscheid wordt gemaakt tussen verschillende partneragents onderling, maar dat er een variabele wordt geleerd die een representatie vormt van vertrouwen versus winstgevendheid in alle partners samen. Dit is niet voldoende voor een cognitief plausibele agent, die moet immers onderscheid kunnen maken tussen ervaringen behorende bij transacties met verschillende handelspartners.

## 4.2 Het matchingsalgoritme

Met behulp van de Cobb-Douglas formule creëren de agents een voorkeursvector, die aan een matchingsalgoritme wordt gegeven. Op basis van de verschillende voorkeursreeksen die de agents opleveren, berekent dit algoritme de bij elkaar passende koppels. Het algoritme is gebaseerd op Tesfatsion's 'deferred choice and refusal' (DCR) algoritme. Het is echter zo gebouwd dat het overweg kan met een aantal uit het model voortvloeiende eisen. Bijvoorbeeld, kopers voegen zichzelf toe aan hun lijstje van leveranciers, zodat ze aan zichzelf kunnen worden gekoppeld.

Op het niveau van het bedrijf kan een koper ervoor kiezen de functies van een leverancier zelf te vervullen. Misschien niet met dezelfde efficiëntie als een gespecialiseerde leverancier, maar wel met de zekerheid goed gekoppeld te worden. Hoewel kopers wel voor zichzelf componenten mogen produceren, mogen ze niet aan andere kopers leveren. Daartegenover mogen leveranciers geen eindproducten produceren. Ten slotte kennen alle agents een quota, een maximaal aantal te leveren of accepteren eenheden en dus ook een maximaal aantal partners waaraan ze kunnen worden gekoppeld.

Het matchingsalgoritme<sup>8</sup> ziet er nu als volgt uit. Alle kopers hebben een 'offer quota' ( $oq$ ), een maximaal aantal leveranciers waarmee mag worden samengewerkt. En alle leveranciers hebben een 'acceptance quota' ( $aq$ ), de maximale hoeveelheid kopers waar aan mag worden geleverd. Alle agents creëren een strikt geordende voorkeursvector van potentiële partners. Wanneer al deze vectoren bekend zijn begint het algoritme<sup>9</sup>:

1. In de eerste stap sturen alle kopers maximaal  $oq$  aanbiedingen naar de meest gewilde leveranciers in hun voorkeursreeks.
2. de leveranciers accepteren een maximum van  $aq$  aanbiedingen van meest gewaardeerde kopers in hun voorkeursreeks.

---

<sup>7</sup> $\tau$  Kent helemaal geen index en is dus voor alle partners gelijk.

<sup>8</sup>Kritiek op het algoritme is, dat het in het voordeel werkt van de kopers, omdat dezen de aanbiedingen versturen. De leveranciers kunnen als het ware niet zelf de markt verkennen en relaties initialiseren.

<sup>9</sup>De eis dat alle vectoren uitgerekend moeten zijn voor aanvang van het DCR algoritme, zal zoals we later zullen zien een aantal gevolgen hebben voor het ACT-R model.



3. Elke koper die geweigerd wordt stuurt voor elke weigering een aanbieding naar de eerstvolgende nieuwe leverancier in zijn voorkeursreeks zolang hier nog leveranciers in voorkomen.
4. de leveranciers kijken opnieuw naar alle aanbiedingen die ze tot nu toe hebben ontvangen en accepteren een maximum van  $aq$  aanbiedingen van meest gewaardeerde kopers in de voorkeursreeks. Zolang er kopers worden geweigerd springt het algoritme terug naar stap 3.

Koper	Leverancier				
	0	1	2	3	4
1	5	1,2	4,4	3,3	2,1
2	2	3,3	4,2	1,4	5,4
3	3	2,4	5,1	1,1	4,2
4	5	4,1	3,3	1,2	2,3

Tabel 4.1: Een voorbeeld van een set voorkeursvectoren. Koper 1 waardeert leverancier 1 als beste, daarna 4, 3, 2 en 0. Leverancier 1 waardeert koper 4 als beste, daarna 1, 2 en 3. Etc. Leverancier 0 staat voor de keuze van een koper om met zichzelf een relatie aan te gaan.

Als we het voorbeeld gegeven in tabel 4.2 uitwerken met voor  $oq = 1$  en  $aq = 2$ , ziet dit er als volgt uit:

1. Kopers 1, 2, 3, 4 versturen 1 verzoek om samenwerking aan respectievelijk leveranciers 1, 3, 3, 3.
2. Leverancier 3 mag maar 2 verzoeken aannemen en wijst koper 2, de koper met de slechtste waardering, af.
3. Koper 2 stuurt een verzoek naar leverancier 0, zichzelf.
4. Dit verzoek wordt ingewilligd.

Het algoritme heeft geleid tot de koper-leverancier paren  $(1, 1)$ ,  $(2, 0)$ ,  $(3, 3)$ ,  $(4, 3)$ , overeenkomstig de situatie weergegeven in Figuur 4.1.<sup>10</sup>

### 4.3 Aanpassingen van het model

De keuze van Klos om de Cobb-Douglas vergelijking te gebruiken bij het bepalen van preferentiescores en de keuze om deze in de vorm van voorkeursvectoren mee

<sup>10</sup>Het algoritme laat de situatie toe waarin een koper aan zichzelf en aan een of meer leveranciers kan worden gekoppeld. Dit kan gebeuren wanneer het 'offer quotum' groter dan 1 is. In deze situatie kan het strikte onderscheid tussen de keuze voor insourcing of voor outsourcing niet meer gehandhaafd worden.

te geven aan het DCR matchingsalgoritme, leiden niet tot de meest cognitief plausible oplossing denkbaar voor een agent. Een betere oplossing zou zijn een agent te maken die op basis van zijn ervaringen in het verleden zelf actie onderneemt om relaties met anderen te initialiseren. Een eerste stap in die richting is de voorkeursvectoren niet te berekenen op basis van de Cobb-Douglas formule, maar dit te laten doen door een cognitief agentmodel volgens de theorie van ACT-R. Het DCR algoritme blijft dan (voorlopig) nog zijn functie behouden. In dit werk beperken we ons tot het zetten van deze eerste stap.<sup>11</sup> Met ACT-R, een architectuur voor cognitie, hopen we een cognitief plausible basis te geven aan onze aangepaste actoren. In het volgende hoofdstuk zullen we het ACT-R agentmodel beschrijven.

---

<sup>11</sup>Een tweede stap zal zijn het vervangen van het DCR matchingsalgoritme. Dit zou goed mogelijk zijn door een nog uitgebreider agentmodel in ACT-R te bouwen, dat ook de capaciteiten krijgt aangemeten waarmee hij zich zo gezegd zelfstandig op de markt zou kunnen begeven. Dit houdt o.a. in de vaardigheden om contacten te leggen en relaties te onderhouden. Met deze eventuele uitbreidingen in het achterhoofd is het verstandig om het gebruik van de complete ACT-R theorie in de simulatie mogelijk te maken. Het is deze overweging geweest, die ons voornamelijk heeft doen kiezen voor het omzetten van ACT-R naar een Object Georiënteerde programmeertaal.

## Hoofdstuk 5

# Het Agentmodel

Dit hoofdstuk beginnen we met een korte uitleg van de belangrijke punten van ACT-R met betrekking tot het model. Zie bijlage C, voor een korte inleiding van de ACT-R theorie.

### 5.1 ACT-R eisen

ACT-R is een prachtig stuk gereedschap om met cognitieve modellen te experimenteren. De theorie is genoeg uitgewerkt om een heleboel beslissingen uit handen van de onderzoeker te nemen. Hoewel ACT-R ervan uitgaat dat kennis in een actor opgeslagen is in de vorm van chunks en productieregels, laat zij nog een heleboel modellering over wat betreft de inhoud van deze twee. Een belangrijke taak die is weggelegd is het kiezen van een geschikte vorm en inhoud van het geheugen. ACT-R kan geen waardeoordeel vellen over de inhoud van het geheugen. Van belang is of een kenniselement nut heeft voor het cognitieve systeem, niet of het al dan niet waar is.

Deze op inhoud gerichte modellering is erg relevant voor vertrouwen (en opportunisme en loyaliteit). Vertrouwen is onderdeel van cognitie maar het is niet gemakkelijk cognitief te doorgronden of te formaliseren. Dit is duidelijk nog een relatief nieuw onderzoeksterrein. In veel gevallen worden ACT-R modellen gespiegeld aan onderzoeksdata. Wij hebben zulke data niet tot onze beschikking. Mede daarom is het verstandig niet al te grote sprongen te maken bij het vormgeven, maar zoveel mogelijk aan te sluiten en gebruik te maken van de mechanismen waarvoor ACT-R bekend staat. Dit betekent in dit onderzoek dat de activatie van chunks veel van het gedrag verklaart. Dit betekent ook dat we geen ingewikkelde zoekstrategieën hebben geformuleerd, maar gebruik maken van *satisficing* en *instance-based learning*. Alle twee deze begrippen komen overeen met het begrip beperkte rationaliteit.

Een andere reden om deze minimale stijl van modelleren toe te passen, heeft te maken met de definitie van vertrouwen. Vertrouwen is gebaseerd op persoonlijke ervaringen en interacties, kortom op informatie die een actor gedurende zijn bestaan verzameld. Vertrouwen is een overtuiging die geïnterpreteerd wordt op basis van deze informatie. Vertrouwen heeft met andere woorden emergente eigenschappen, of om in Dennett's (1978, 1981) termen te praten, intentionele eigenschappen.

Belangrijk is het onderscheid dat ACT-R maakt tussen procedureel en declaratief geheugen. In het declaratief geheugen zit kennis waar we vrijelijk toegang tot hebben en waar we ons bewust van zijn. Procedurele, onbewuste kennis komt meer tot uitdrukking in ons gedrag. Procedurele kennis beschrijft meestal hoe we declaratieve kennis moeten gebruiken bij het oplossen van problemen.

Verder is er nog het onderscheid tussen symbolische en subsymbolische kennis. Het symbolische niveau zijn de goals, procedures en chunks die op een semantisch niveau interacteren. Het subsymbolische niveau wordt gevormd door een verzameling variabelen, die de snelheid en sterkte van procedures en chunks bepalen. Dit niveau komt overeen met neuronale activatie processen.

Tenslotte nog deze vraag, hoe vergaart een agent nieuwe kennis? Er zijn in principe drie vormen van symbolisch leren. Ten eerste het toevoegen van nieuwe chunks aan het declaratief geheugen is wat ACT-R's perceptuele component doet wanneer aandacht gericht wordt op objecten in de buitenwereld. Er worden dan representaties toegevoegd in de vorm van chunks. Een tweede manier van chunkvorming is wanneer een procedure 'popt', wat zoveel wil zeggen dat als de goal opgelost is, het doel is bereikt. Er wordt dan een chunk gemaakt die een representatie is van een oplossing voor een bepaald probleem. De derde manier is 'production compilation', dat wil zeggen het vormen van nieuwe procedures. Deze scriptie beperkt zich tot de hierboven als tweede genoemde leermethode van ACT-R, zoals zal blijken uit het in dit hoofdstuk te behandelen agentmodel.

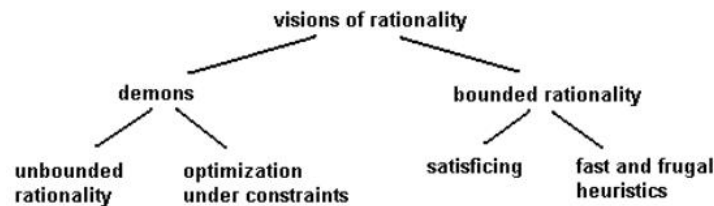
Niet geheel verassend is er ook nog zoets als subsymbolisch leren, dit houdt in dat de sterktes van associaties en procedures en de activaties van chunks aangepast worden. Deze subsymbolische parameters zijn schattingen van kansen op gebeurtenissen en kosten van acties, die met iedere nieuwe handeling worden bijgesteld. Dit is een automatisch proces van aanpassing in ACT-R. Het resultaat van iedere cognitieve handeling wordt teruggekoppeld aan de subsymbolische parameters. Je zou kunnen spreken van optimalisatie of stochastisch leren. Nuttige, veel gebruikte associaties worden op die manier versterkt, ten opzichte van minder belangrijke relaties.

In dit hoofdstuk wordt verder ingegaan op de begrippen *satisficing* en *instance-based-learning*. Daarna wordt de implementatie van het agentmodel behandeld.

## 5.2 Satisficing

Hoe kan iemand rationeel handelen in een wereld waarin kennis en tijd beperkt zijn en goed en lang nadenken vaak een luxe is? Traditionele modellen van onbeperkte rationaliteit en optimalisatie zoals die in de cognitiewetenschap, economie, en biologie voorkomen, waren geneigd beslissingsmakers te zien alsof ze bovennatuurlijke rationale vermogens, onbeperkte kennis, en oneindig veel tijd bezaten<sup>1</sup>. Maar beslissingsprocessen hebben een meer psychologisch geloofwaardige benadering nodig, waarbij alle drie de capaciteiten van een actor worden beperkt (Todd en Gigerenzer, 1999). (1) Ze moeten simpel zijn wat betreft de gebruikte algoritmen. (2) Ze moeten voldoende robuust zijn om met beperkte informatie te kunnen werken. Informatie die ook nog eens van heel uiteenlopende aard en daarom moeilijk te vergelijken kan zijn. En ze moeten (3) snel zijn in de zin van dat er weinig berekeningen nodig zijn om tot een beslissing te komen.

Met al deze beperkingen zou je misschien niet meer verwachten dat de beslissingsprocessen nog tot redelijke resultaten kunnen leiden. Echter dit lijkt alleszins mee te vallen. Beperkte processen kunnen net zo accuraat zijn als processen die alle registers opentrekken (Todd en Gigerenzer, 2000). Op basis van de bovengenoemde drie eigenschappen van rationaliteit onderscheiden Todd en Gigerenzer vier voorkomens van rationaliteit, waarbij achtereenvolgens tijd, kennis en algoritmische complexiteit worden beperkt.



Figuur 5.1: Visions of rationality, uit 'Behavioral and Brain Sciences'(2000) 23, p. 729.

De geestelijke vader van beperkte rationaliteit, Herbert Simon, zag twee componenten van waaruit hij zijn visie vormde. De beperkingen van de menselijke geest en de structuur van de omgeving waarin deze zich bevindt (Simon, 1956a). Vanwege beperkte geestelijke capaciteiten moeten mensen gebruik maken van bij benadering gekozen methoden. Waarom deze benaderingen toch nog zo goed werken, heeft alles te maken met de structuur van de omgeving. Immers als de omgeving goed te vangen is in een abstractie, is het voor een simpele heuristiek

<sup>1</sup>Wat door sommigen, zoals Sargent (1993), als argument wordt gebruikt voor het gebruik van onbeperkte rationaliteit tegenover beperkte rationaliteit in modellen, is dat de eerste aanpak minder parameters met zich mee zou brengen en mathematisch minder complex zou zijn.

mogelijk zich hieraan goed aan te passen. Dit is wat Gigerenzer en Todd ecologische rationaliteit noemen (Todd en Gigerenzer, 2000).

TCE gaat ook uit van een beperkt rationele actor. Maar waar TCE het liet bij dit uitgangspunt in de veronderstelling, dat het uiteindelijke resultaat niet verschilt of er nu sprake is van beperkt of onbeperkt rationele actoren, wordt dit hier verder uitgediept.

Een belangrijke vorm van beperkte rationaliteit is satisficing<sup>2</sup> (Simon, 1958). Dit is een methode voor het maken van een keuze uit een aantal alternatieven. Satisficing houdt op met het zoeken als een gevonden alternatief eenmaal boven een bepaalde waarderingsdrempel komt. Anders gezegd, de actor kijkt niet verder als hij tevreden is met een gevonden oplossing. Nog steeds moet bij satisficing echter opgelet worden dat de beperkte rationaliteit niet uit het oog wordt verloren. Het zou bijvoorbeeld fout zijn om een drempelwaarde te kiezen die op zichzelf moeilijk te berekenen zou zijn of moeilijk te hanteren criteria gebruikt voor een vergelijking van de alternatieven.

### 5.3 'Vertrouwen' in de simulatie

In deze simulatie draait het telkens om het gekoppeld worden aan een zo goed mogelijke partner. Vertrouwen wordt gedefiniëerd als het voortzetten van een relatie. Dat wil zeggen, de agent vormt een relatie met dezelfde partner als in de vorige ronde. Het complement hiervan heet opportunisme, een agent verbreekt de relatie. De verwachting is dat een agent die geneigd is de voorkeur te geven aan het voortzetten van een relatie, op de lange duur leidt tot meer winstgevende transacties, dan wanneer een agent door opportunistisch gedrag niet toekomt aan het vormen van langdurige relaties.

Competentie speelt in deze simulatie geen rol. Een agent is namelijk altijd in staat zijn verplichtingen tegenover een ander na te komen; Hij zal altijd de gevraagde componenten kunnen leveren of het eindproduct kunnen verkopen. Waar het om gaat is of de andere agent de intentie heeft het verzoek tot samenwerking positief te beantwoorden. Het leren of een actor betrouwbaar is, geschikt om een lange relatie mee aan te gaan, komt overeen met het maken van een representatie hiervan.

Vertrouwen ligt opgesloten in ervaringen die een agent in de loop der tijd opdoet. Vertrouwen is een verschijnsel dat de agent interpreteert uit zijn ervaringen. Vertrouwen is een impliciet begrip dat emergente eigenschappen kent. Een agent mag niet een variabele bijhouden voor het vertrouwen in een bepaalde agent. Vertrouwen moet blijken uit gedrag dat gebaseerd is op zijn ervaringen. Daarom de keuze voor een simpele vorm van leren, die heel weinig aannames maakt als het gaat om wat er nodig is om vertrouwen te leren.

---

<sup>2</sup>Satisficing is een samentrekking van de termen sacrificing en satisfying. En houdt zoveel in als tevreden zijn met een gevonden keuze, terwijl potentieel betere nog onbekende keuzes worden opgeofferd.

## 5.4 Instance-Based Learning

De agents moeten episodes, op zichzelf staande handelingen, onthouden, die telkens hun eigen unieke input en output waarden hebben. Deze input en output representeren zo getrouw mogelijk de input en output waarmee een agent in een bepaalde simulatieronde wordt geconfronteerd. Dit betekent dat er geen expliciete informatie in de vorm van een variabele over bijvoorbeeld vertrouwen wordt opgeslagen, omdat hij hierover ook geen expliciete informatie ontvangt. Elke afgeronde transactie leidt tot een episode-chunk in het geheugen van de agent. De inhoud van deze chunk wordt bepaald door de identiteit van de partner en de behaalde winst.

Bovenstaande methode komt grotendeels overeen met *instance-based learning* (Aha en Kibler, 1991). De simpelste vorm van leren is het vormen van episodes om ervaringen in op te slaan in het geheugen. Wanneer een object is waargenomen of de oplossing van een probleem is gevonden wordt dit opgeslagen in het geheugen. Het geheugen kan gezien worden als een grote opzoektafel. Wanneer men een volgende keer een nieuw object of probleem tegenkomt wordt het geheugen doorgelopen op zoek naar een vergelijkbaar voorval. Het voorval hoeft hierbij natuurlijk niet precies te passen al naar gelang de eisen die je wil stellen aan het algoritme. Hierdoor kan een mate van generalisatie plaatsvinden.

Dit alles heeft grote gevolgen voor het uitlezen van de resultaten in de experimenten. Omdat vertrouwen nu niet meer af te lezen is zoals de snelheid op een kilometerteller<sup>3</sup>, moet vertrouwen geïnterpreteerd worden kijkende naar de gevormde relaties of de activiteiten in ACT-R.

## 5.5 Aanpassingen op het vorige ACTCE-model

In het model van Klos worden begrippen als vertrouwen en loyaliteit gevangen in een variabele. Deze score werd berekend door de Cobb-Douglas formule (zie formule 4.9). De rangschikking van de voorkeuren voor bepaalde partners werd hierdoor bepaald. Deze score was een functie van de verwachte winst, het vertrouwen en de loyaliteit. Het invoegen van een variabele voor vertrouwen, zorgt ervoor dat de score niet slechts afhangt van de verwachte winst. In het huidige model worden deze begrippen niet expliciet gerepresenteerd door middel van variabelen, maar drijven impliciet boven op het gedrag van de agent zoals dat ontstaat uit zijn cognitieve structuur. De agents kunnen uiteraard wel van binnen bekeken worden.

Het punt waar het model van Klos aangepast wordt is de Cobb-Douglas formule. Deze formule is niet meer nodig voor het bepalen van de voorkeursvectoren. De agents zullen nog steeds een afweging tussen opportunisme en vertrouwen kunnen maken. Vertrouwen is echter niet meer af te lezen uit een variabele. Er

<sup>3</sup>In het model van Klos kan gekeken worden naar de variabele  $t$  om het vertrouwen af te lezen.

moet nu gekeken worden naar de activatie van chunks, de gevormde koppels en de voorkeuren. Klos maakte in zijn model gebruik van 'reinforcement learning' waarbij agents leren om de uitkomst van een payoff-functie te maximaliseren. De variabelen  $\alpha$  en  $\tau$ , die in het model van Klos zorgen voor adaptatie, hebben plaatsgemaakt voor het adaptieve karakter van ACT-R zelf.

## 5.6 Een ACT-R agentmodel

Wat moet het ACT-R model doen? Elke ronde slaat een agent informatie over de behaalde winst en de identiteit van de partner op in de vorm van chunks in zijn geheugen. De chunks in het geheugen zijn allemaal op zichzelf staande ervaringen, zoals bij instance-based learning. Uit de groeiende verzameling van chunks kiest de agent telkens chunks en selecteert op die manier een partner. Net als bij satisficing wordt gebruik gemaakt van drempelwaarden om alle mogelijkheden op te selecteren. Dit betekent dat als de winst onder een bepaalde waarde ligt, de chunk niet uit het geheugen wordt opgehaald. Als wel aan deze eisen is voldaan, dingen ze niet meer mee op inhoud maar op basis van hun activatiewaarde in het geheugen. De volgorde waarin de chunks worden gekozen, bepaalt hun plaats in de voorkeursrangschikking.

In de meeste gevallen zal dit leiden tot een conservatieve keuze. Chunks die worden geretrieved worden immers versterkt in activatie, waardoor de kans om de volgende keer te worden geretrieved groter wordt. Dit conservatisme leidt tot vertrouwensvol gedrag. Opportunistisch gedrag zal veroorzaakt worden doordat een groep chunks met hoge activatie onderling gaat concurreren om uitgekozen te worden<sup>4</sup>.

Een chunk in het model ziet er als volgt uit,

ACT-R		Beschrijving
geheugenChunk:		Een chunk heeft een naam nodig.
isa	episode	Een opgedane ervaring
name	'Agent 1'	met Agent 1
profit	0.20	waarbij 0.20 winst werd behaald.

Tabel 5.1: Een geheugenchunk in het model.

Er is een populatie van agents. Elk van de agents representeert hierin een bedrijf. De agents hebben een aantal representaties van andere agents, die ze aan het begin van de simulatie krijgen. Deze representaties hebben allemaal een variabele activatiewaarde. In de loop van de simulatie leert een agent door het veranderen van de activatiewaardes welke partner hij moet kiezen. Verder

<sup>4</sup>Chunks moeten eerst onderling concurreren in het geheugen van de agent om terecht te mogen komen in de voorkeursvector. Daarna moeten ze door het matchingsalgoritme geselecteerd worden als handelspartner.



kunnen ze beslissen op basis van een winst, die een indicatie vormt van de winst die ze zullen maken wanneer ze met een bepaalde partner gaan samenwerken. De claim is niet dat agents vertrouwen moeten hebben, maar wel dat het mogelijk voor ze is om te kiezen voor een partner van wie de winst niet noodzakelijk de hoogste is.

Wat voor ACT-R chunks en procedures zijn er nodig? Een voorbeeld van de gebruikte chunks was hierboven te zien. Hieronder staat één van de productieregels van het model<sup>5</sup>.

ACT-R	Beschrijving
(P chooseAgent =goal> isa    episode first  nil ==> +retrieval> isa    episode name   =name > profit  0.5 =goal> first  =name )	De naam van de productie. Als de goal een episode is en er nog geen agent is gevonden Dan haal een chunk uit het geheugen van het type episode onthoud de naam van de agent als de winst meer dan 0.5 is verander de goal door de naam in te vullen

Tabel 5.2: Een productieregel uit het model.

De verwachting is dat aan het gedrag van het model en de inhoud en activatie van de chunks te zien is of een agent vertrouwen heeft in een andere agent of niet. Vertrouwen had ook op een andere manier kunnen worden geïmplementeerd, namelijk als een speciale chunk voor vertrouwen in het geheugen die bijhoudt in hoeverre een bepaalde agent te vertrouwen is. In verband met de keuze voor het leren van input-output episoden, valt vertrouwen in de vorm van een speciale chunks af. Het komt erop neer dat de vertaalslag van gedrag naar vertrouwen niet aan het begin, tijdens de modellering van het model, maar aan het eind wordt gemaakt, bij de interpretatie van de resultaten. Dit is psychologisch plausibeler, omdat ervaringen op deze manier direct het gedrag bepalen, zonder een extra stap te hoeven maken via door het geheugen bij te houden kennis over vertrouwen tijdens iedere opgedane ervaring.

Aan het eind van een simulatieronde krijgen de agents feedback in de vorm van hoeveel winst ze behaald hebben met welke partner. In het geval van samenwerking delen kopers de winst gemaakt op het product met de leverancier van de componenten.

<sup>5</sup>Het model gebruikt drie productieregels, om de voorkeursvector te vullen met drie uitgekozen chunks. Verder is er nog een productie om deze vector te communiceren naar het matchingsalgoritme. Zie bijlage A om een overzicht te krijgen van het gehele model inclusief failproducties en chunks.

De kopers wordt het mogelijk gemaakt een voorkeursvector van leveranciers te genereren, waarin ze zelf ook gerangschikt zijn. In het geval kopers zichzelf de hoogste voorkeur geven komt dit overeen met de beslissing zelf de componenten te produceren<sup>6</sup>. De leveranciers genereren op hun beurt een voorkeursvector van kopers. Zelf komen ze niet in deze vector voor omdat dit geen betekenis heeft. Een leverancier kan binnen de simulatie namelijk nooit direct op de final-goods markt leveren.

De voorkeuren worden door een matchingsalgoritme verwerkt tot een serie koppelingen tussen kopers en leveranciers. Het risico dat agents lopen bij het vormen van een voorkeursvector en daarna bij het overhandigen aan een matchingsalgoritme, is het gekoppeld worden aan een andere dan hun eerste keus. Dit is iets wat vooral zal voorkomen als een bepaalde agent door veel anderen erg hoog wordt gewaardeerd. De kans door een felbegeerde agent afgewezen te worden is natuurlijk groter dan afgewezen te worden door een minder felbegeerde agent. Vaak zal het dus slimmer zijn om de voorkeur te geven aan minder gewilde partners en om een langdurige relatie op te bouwen met een vaste partner. Te verwachten valt dat er een initiële aftastproces zal plaatsvinden tussen agents. Uiteindelijk kan dan een stabiele situatie optreden, waarbij agents de voorkeur leren te geven aan anderen met een wederzijdse voorkeur. Deze verwachting van langdurige wederzijdse relaties is te interpreteren als een vorm van vertrouwen in een relatie, dat in de loop van de simulatie is gegroeid.

In de simulatie kunnen zich economische leereffecten alleen voordoen als een relatie zonder onderbreking over meerdere tijdstappen wordt voortgezet. Trouw aan de huidige partner wordt dus vanwege economische redenen beloond. Hierdoor, en omdat een actor alleen informatie vergaart over agents wanneer hij met ze samenwerkt, ontstaat er een spanningsveld. Aan de ene kant wil een agent op zoek gaan naar betere partners en nieuwe informatie verzamelen. En aan de andere kant wil hij zoveel mogelijk voordeel halen uit een langdurige relatie en geen winst meer kwijtraken aan het zoeken naar andere actoren. Dit spanningsveld wordt ook wel het exploratieprobleem genoemd en de tegenstrijdige belangen van exploratie versus exploitatie moeten tegen elkaar worden afgewogen (Russel en Norvig, 1995, p. 609).

Samenvattend ziet nu een mogelijke agent in ACT-R, die in staat is om een voorkeur te bepalen, er zo uit:

1. initiële doel: vindt voorkeurspartner.
2. Voer een procedure uit om een chunk/partner uit het geheugen te halen.
3. Herhaal de vorige stap om meer kandidaten in de voorkeursvector te krijgen.
4. Geef deze kandidaten in de volgorde van ophalen door aan het Matchingsalgoritme.

---

<sup>6</sup>Dit is inherent aan de werking van het matchingsalgoritme. Zodra een koper een verzoek tot samenwerking aan zichzelf doet zal hij zijn eigen verzoek inwilligen.

5. Wacht op feedback over het uiteindelijke resultaat.
6. Pop de goal waardoor de feedback wordt opgeslagen.
7. En begin aan de volgende ronde.

Wanneer een heleboel van deze agents bij elkaar wordt gezet in een omgeving, hopen we te zien dat ze op zoek gaan naar agents die ze kunnen vertrouwen en daarmee langdurige relaties opbouwen.

## 5.7 ACT-R in een ACTCE model

De twee grote nadelen van het gebruik van ACT-R in een Multi-Agent context zijn in mijn ogen het ontbreken van een Multi-agent ondersteuning en de moeilijkheid waarmee ACT-R gekoppeld kan worden aan andere systemen. Dit kunnen niet cognitieve modellen zijn maar ook cognitieve uitbreidingen geschreven in een andere omgeving dan ACT-R<sup>7</sup>. Er is een gebrek aan aandacht binnen ACT-R voor beide punten.

Om de hierboven beschreven agent in het ACTCE model te passen is het nodig de ACT-R architectuur op een aantal punten aan te passen. Allereerst moet de architectuur uitgebreid worden met de mogelijkheid om te communiceren met de simulatie daarbuiten, liefst op een directe en simpele manier. Ten tweede is ACT-R niet in zijn geheel nodig, vanwege de eenvoud van het ACT-R model. Ten derde wordt er rekening gehouden met het feit dat in de toekomst ACT-R wel in zijn geheel geïmplementeerd moet worden om gebruik te maken van andere modellen. Ten slotte moet de architectuur geschikt gemaakt worden voor het gebruik in een Multi-Agent Systeem, wat bijvoorbeeld betekent dat een model stopgezet kan worden op elk gewenst moment tijdens zijn uitvoering en kan communiceren met andere simulatiemodules.

Het stopzetten van het ACT-R model is nodig vanwege het uitrekenen van de koppelingen door het DCR matchingsalgoritme. Het DCR moet wachten op alle agents om hun voorkeuren te berekenen. Het DCR kan immers pas uitgevoerd worden als alle voorkeursvectoren bekend zijn. En de agent kan zijn cyclus pas afronden als het DCR uitgerekend heeft wie zijn partner is geworden. Het DCR zweeft in dit opzicht als een soort van marktmeester boven de markt.

De agent ontvangt feedback over aan wie hij gekoppeld is en wat de behaalde winst is. Deze informatie wordt door de agent opgeslagen in een nieuwe chunk. In principe worden al deze chunks opgeslagen als verschillende episodes, ook die van vorige keren. Op die manier verzamelt de agent echter een enorm aantal chunks als de simulatie voor langere tijd draait. Omdat veel van deze chunks een vergelijkbare inhoud kennen, is gekozen voor het implementeren van een

---

<sup>7</sup>Er is sinds kort een versie van ACT-R geschreven in Java uitgebracht op <http://jactr.sourceforge.net/>.

generalisatieproces in het geheugen<sup>8</sup>. Dit komt erop neer dat de gemaakte winst wordt afgerond. En chunks met gelijke naam en gemaakte winst worden samengevoegd, in overeenstemming met de theorie van ACT-R, waardoor hun activatie toeneemt.

De formules die nodig zijn voor het eerste ACT-R model zoals hierboven beschreven zijn de activatievergelijking in afgeslankte vorm<sup>9</sup> en de benaderde base-level activatie formule.

$$A_i = B_i + \sigma \quad (5.1)$$

$$B_i \approx \log \left( \frac{n}{1-d} \right) - d \log L \quad (5.2)$$

Waarom slechts deze twee worden gebruikt heeft te maken met het feit dat productieregels niet met elkaar hoeven te concurreren. Er is namelijk nooit meer dan één regel van toepassing op de goal.

Het resultaat van al dit programmeerwerk is te zien in bijlage B, die een overzicht geeft van alle classes en hun onderlinge relaties in een objectmodel en tevens een scenarioview. Dit sluit al het voorbereidende werk af om de simulatie te kunnen laten draaien.

---

<sup>8</sup>De feedback over de behaalde winst verschilt vaak minimaal per ronde. Omdat de inhoud telkens net iets anders is beschouwt ACT-R ze als verschillende chunks en creëert telkens een nieuwe chunk in het geheugen.

<sup>9</sup>Retrievalspecificaties en contextactivatie vanuit de goal spelen in het model geen rol.

# Hoofdstuk 6

## Resultaten

### 6.1 Een TCE experiment

Een eerste test die gedaan wordt met ons model is om te kijken of de uitkomsten overeenkomen met TCE voorspellingen. Dit heeft betrekking op de relatie tussen toenemende produktspecificiteit en meer insourcing versus outsourcing. Er wordt gekeken of hogere differentiaties ook vaker leiden tot kopers die zichzelf als partner hebben, zoals TCE voorspelt.

De parameters die voor deze eerste simulatie worden gebruikt zijn de volgende:

	Parameter	Waarde	Opmerkingen
<b>ACTCE</b>	Aantal kopers	5	in de simulatie
	Aantal leveranciers	5	idem
	Aantal tijdstappen	100	duur van 1 simulatie
	Aantal simulaties	100	waarover gemiddeld wordt
	Offerquotum koper	1	koper kent max. 1 partner
	Acceptancequotum leverancier	3	leverancier kent max. 3 partners
	Differentiaties	{0.5 , 0.6}	en dit alles voor 2 differentiaties
<b>ACT-R</b>	Base-level learning	0.5	defaultwaarde in ACT-R
	Ruisvariantie	0.5	idem
	Leeftijd van initiële chunks	0	idem
	Aantal presentaties van initiële chunks	1	idem
<b>Agentmodel</b>	Afrondingsfactor winst	0.05	hierop wordt afgerond
	Zelfrepresentatie koper	true	vanaf het begin van de sim.
	Bovengrens voor winst initiële chunks	0.5	niet hoger
	Ondergrens voor winst initiële chunks	0.2	niet lager
	Eerste satisfyingdrempel voor winst	0.0	het minimum voor winst-slot
	Tweede satisfyingdrempel voor winst	0.0	idem, maar nu voor tweede keus
Derde satisfyingdrempel voor winst	0.0	idem, maar nu voor derde keus	

Tabel 6.1: Parameters voor experiment 1, behorende bij figuren 6.1 t/m 6.6.

De parameters zijn in drie groepen te verdelen. Allereerst zijn er de parameters

die iets zeggen over het ACTCE model, het aantal kopers, leveranciers, tijdstappen, en het aantal simulaties waar het resultaat over gemiddeld moet worden. Bij de keuze voor het aantal agents moet rekening worden gehouden met de volgende afweging. Aan de ene kant wanneer dit aantal te hoog is, wordt de kans op een wederzijdse keuze van agents voor elkaar sterk verminderd. Er zijn dan immers meer partners om uit te kiezen. Aan de andere kant wanneer dit aantal te laag is, wordt het vinden van een partner te makkelijk. Het proces van zoeken op de markt wordt dan teniet gedaan. De keuze voor het aantal tijdstappen wordt bepaald door de tijd die de simulatie nodig heeft om na het opstarteffect redelijkerwijs tot rust te komen. Het aantal simulaties wordt zo gekozen dat de resultaten duidelijk scheidbaar van elkaar worden. Dit aantal hangt samen met het aantal agents in de simulatie. Hoe meer agents in de simulatie des te minder simulaties er nodig zijn. Verder is er het offerquotum van de koper, dat aangeeft met hoeveel leveranciers hij mag samenwerken. Aanvullend is er het acceptancequotum, dat zegt aan hoeveel kopers een leverancier maximaal mag leveren. Beide waardes zijn hier zo laag mogelijk gehouden, omdat het anders te gemakkelijk wordt voor een koper om een partner te vinden, net als in het geval van een laag aantal agents in de simulatie. Het acceptancequotum is echter iets hoger gekozen met de bedoeling om schaaleffecten mogelijk te maken. En tenslotte de laatste waarde in deze groep, de differentiatie, die in het model als globale variabele wordt ingesteld en waar kopers dus niet zelf een waarde voor mogen kiezen. De differentiatie wordt gevariëerd om TCE te kunnen toetsen.

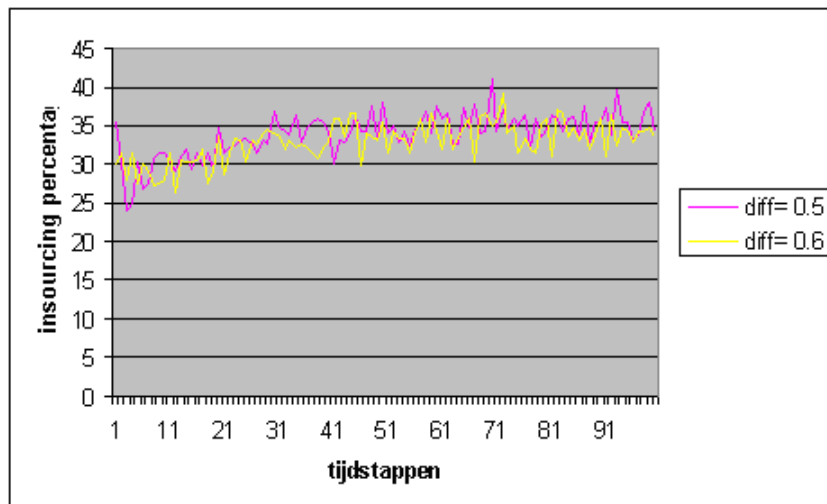
De tweede groep zijn ACT-R parameters. In principe geldt hier dat defaultwaarden van ACT-R gebruikt worden. De base-level learning waarde bepaalt hoe snel een chunk in het geheugen van activatie vermindert, hoe hoger deze variabele des te groter de activatiedaling. De ruisvariantie bepaalt de mate van ruis die bovenop de activatie komt. Hoe groter de variantie, des te groter de ruis. In experiment 3 is te zien wat het effect is wanneer deze waarde gevariëerd wordt. De leeftijd van initiële chunks, de chunks waar een agent de simulatie mee begint, kan gezet worden. Hogere leeftijd betekent een kleinere activatie bij aanvang. Een tegenovergesteld effect heeft het aantal representaties van de chunks voor de simulatie begint. Meer representaties betekent een hogere activatie bij aanvang.

In de derde groep parameters bevinden zich alle variabelen die volgen uit het agentmodel. De afrondingsfactor geeft de eenheden aan waarop de feedback over de behaalde winst wordt afgerond. Als deze factor omhoog gaat, betekent dit minder vorming van nieuwe chunks in het geheugen en meer versterking van bestaande chunks door generalisatie. De gekozen waarde laat voldoende vorming van nieuwe ervaringen toe en houdt tegelijkertijd het aantal nieuwe chunks in het geheugen binnen de perken. De boven- en ondergrens voor de winst van initiële chunks bepalen het minimum en maximum waartussen de winst bij aanvang willekeurig wordt gekozen. Door deze grenzen aan te passen kan de inhoud van een agents geheugen bij aanvang van een simulatie gemanipuleerd worden. Door de grenzen naar beneden bij te stellen, wordt het moeilijker voor een agent om zich een geschikte ervaring te herinneren. Omgekeerd, door ze naar boven bij te

stellen, zijn meer agents mogelijke partners. De drie satisfyingdrempels bepalen de minimale hoeveelheid winst waar een representatie aan moet voldoen wil hij een van de drie keer worden gekozen uit het declaratief geheugen, tijdens het vullen van de voorkeursvector. Waarom deze drie variabelen allemaal op 0.0 staan wordt besproken in de conclusies in het volgende hoofdstuk.

In de figuren 6.1 t/m 6.6 is het resultaat te zien van een simulatie van 5 kopers, die een relatie kunnen aangaan met maximaal 1 leverancier uit een groep van 5. De leveranciers daarentegen kunnen relaties aangaan met 3 kopers op hetzelfde tijdstip. Elke simulatie heeft een duur van 100 tijdstappen. De resultaten zijn gemiddeld over alle 5 kopers samen en 100 simulaties.

In figuur 6.1 geeft elke lijn het percentage aan van de productie die door de koper met zelfgemaakte componenten is vervaardigd. Er is geen relatie te zien tussen insourcing en differentiatie. Dit is anders dan TCE voorspelt. Volgens TCE had de grafiek voor insourcing hoger moeten liggen bij hogere differentiaties. Wat wel zichtbaar is, is dat de proportie insourcing stijgt naarmate de simulatie vordert. In het begin van de simulatie maakt een koper nog weinig onderscheid tussen zichzelf en leveranciers als partner. Pas na verloop van tijd leert hij te kiezen voor zichzelf.

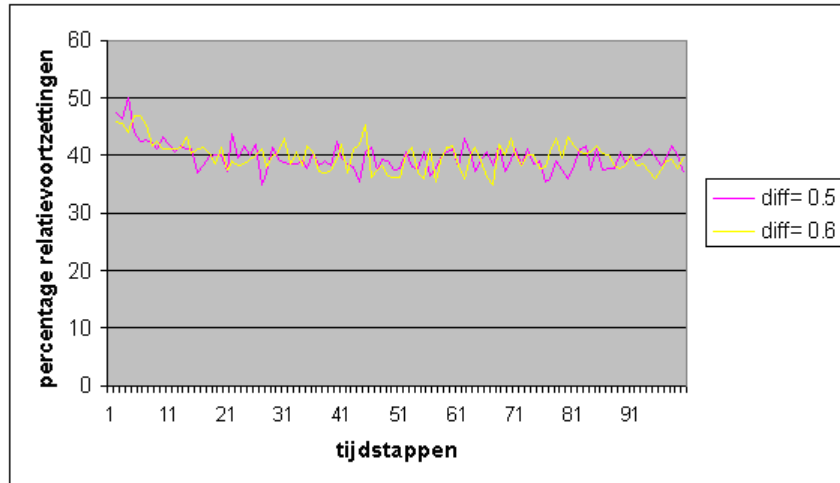


Figuur 6.1: Insourcing percentage voor verschillende differentiaties.

Kijkende naar Figuur 6.2 is te zien dat de koper leert te veranderen van partner naarmate de simulatie vordert. Dit verbreken van relaties heeft in het ergste geval, wanneer een nieuwe leverancier niet wil samenwerken, tot gevolg dat hij alsnog voor zichzelf componenten mag maken<sup>1</sup>. Uit deze resultaten blijkt dat

<sup>1</sup>Zo is immers het matchingsalgoritme vormgegeven, een koper heeft zichzelf om mee te produceren, als geen leverancier dat wil.

de agents weinig langdurige relaties opbouwen. Relaties met leveranciers duren niet lang genoeg om te resulteren in significant hogere opbrengsten. Met andere woorden leereffecten en schaaleffecten leveren in het begin van een relatie niet genoeg op om tot voortzetting van de relatie te komen.



Figuur 6.2: Percentage relatievoortzettingen voor verschillende differentiaties.

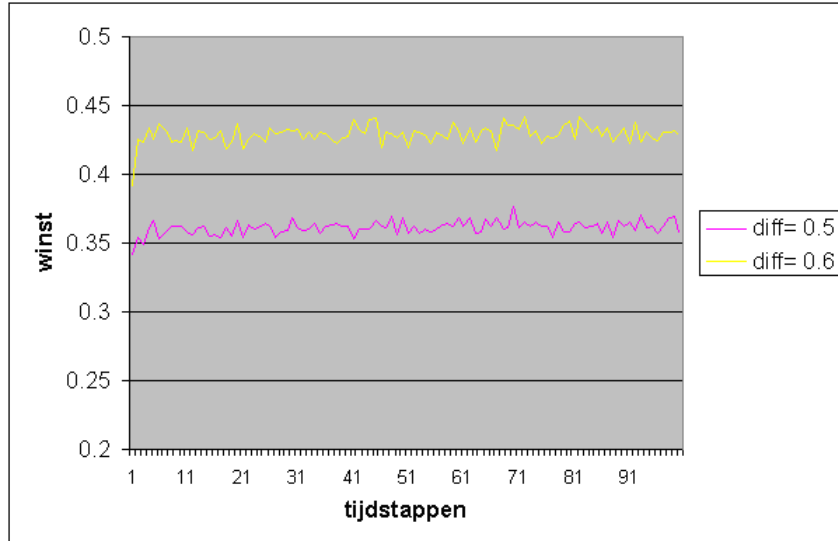
Hier zien we een aspect van de simulatie waarmee in TCE geen rekening wordt gehouden. Het startup effect toont aan dat markten tijd nodig hebben om te worden gevormd. Actoren hebben tijd nodig om te leren welke keuzes ze moeten maken. Niet alleen economische leereffecten hebben tijd nodig om zich te onwikkelen, ook het vormen van stabiele relaties vereist een aftastingsfase van de kant van de agent. En als schaaleffecten willen optreden, zullen deze stabiele relaties enigszins aanwezig moeten zijn.

De winst die kopers maken is wel een functie van differentiatie, zie Figuur 6.3. Hogere differentiatie geeft meer winst. Niet zo gek als je je bedenkt dat ze in dat geval ook hogere productopbrengsten binnenkrijgen. Om voor dit verschijnsel te compenseren moeten we gaan normaliseren.

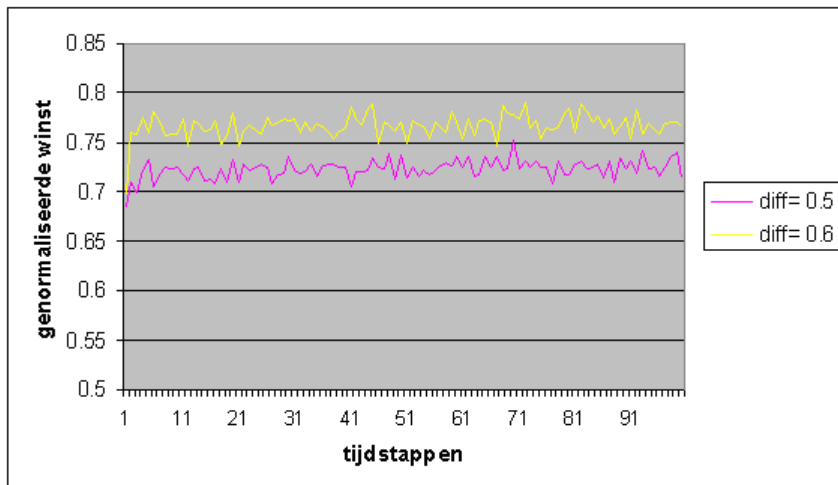
Hiervoor is een maat nodig voor de maximaal te behalen winst in een simulatie. Twee vormen van optimale winst zijn denkbaar. Allereerst de methode die Klos ook gebruikt, de maximaal te halen winst wordt gevormd wanneer schaal- en leereffecten maximaal zijn. Dit betekent dat de koper een relatie moet aangaan met een leverancier, die aan zoveel mogelijk kopers levert om maximale schaaleffecten te behalen. Bovendien moet deze relatie gedurende de gehele simulatie worden onderhouden, voor het bereiken van maximale leereffecten. Het resultaat van normalisatie met deze maat is te zien in figuur 6.4.

De andere vorm van maximale winst is de winst die behaald wordt wanneer een



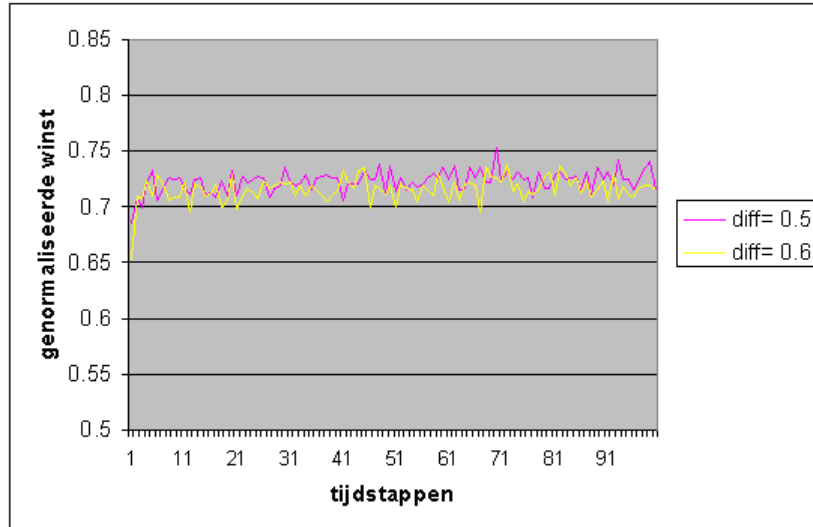


Figuur 6.3: De gemiddeld behaalde winst van een koper.



Figuur 6.4: Genormaliseerd voor een constante relatie met de voordeligste leverancier.

koper altijd voor insourcing kiest. Beide efficiëntie effecten zijn dan wel afwezig, maar de winst hoeft tenminste niet gedeeld te worden. Het gevolg van deze methode is te zien in figuur 6.5. Figuur 6.5 laat duidelijk zien dat met de juiste



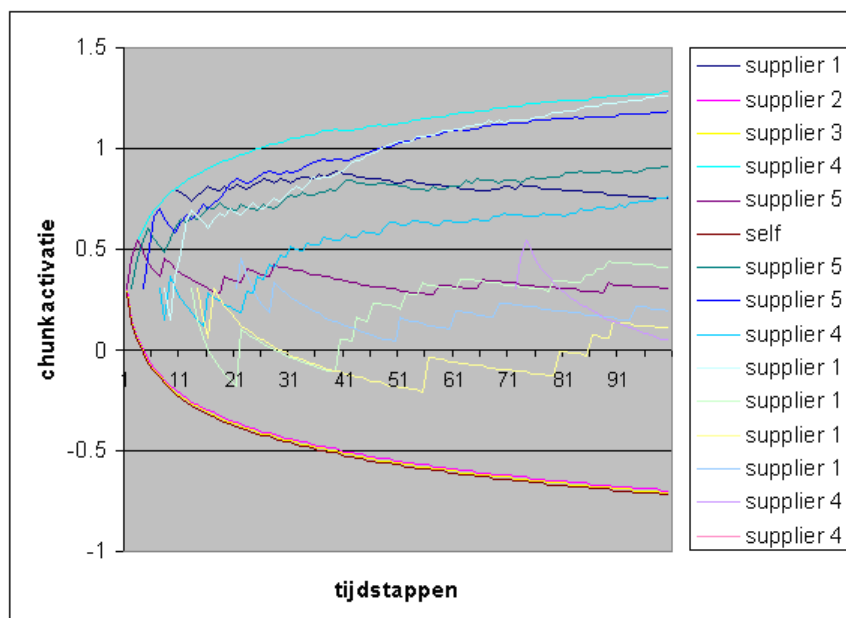
Figuur 6.5: Genormaliseerd voor voortdurende insourcing.

normalisatie het onderscheid tussen de verschillende differentiaties verdwijnt. Beide figuren vergelijkend moet geconcludeerd worden dat de gemaakte winst grotendeels, zo niet geheel, wordt verklaard door het insourcing cijfer en niet door schaal- of leereffecten. Dit staft de eerdere constatering dat langdurige relaties zich niet ontwikkelen. Hierdoor is relatieverbreking of -voortzetting niet gerelateerd aan differentiatie. Beide grafieken laten in ieder geval wel duidelijk zien dat optimale opbrengsten niet worden bereikt. De veronderstelling van TCE dat dit als vanzelfsprekend wel zou gebeuren, wordt hiermee tegengesproken.

## 6.2 Een kijkje in de agent

Er kan in iedere ACT-R agent gekeken worden om te zien hoe chunks met elkaar concurreren om activatie. De gemiddelde resultaten zoals hierboven zijn echter niet noodzakelijk terug te vinden in het gedrag van een enkele agent. De individuele variatie wordt onzichtbaar wanneer gemiddeld wordt over meerdere agents.

Figuur 6.6 toont de activiteiten van de chunks in het declaratief geheugen van één enkele koper uit dezelfde simulatie als hierboven (de differentiatie is in dit geval



Figuur 6.6: Een Type A agent.

0.5). De eerste 6 chunks zijn op tijdstip 0 gevormd zij vormen de zogenaamde initiële chunks. De helft van deze initiële chunks daalt vanaf het begin in activatie. Zij worden niet uit het geheugen opgehaald of versterkt door tijdens de simulatie opgedane nieuwe ervaringen. Onder deze 3 niet gebruikte chunks bevindt zich ook de zelfrepresentatie. Deze koper zal waarschijnlijk weinig aan insourcing hebben gedaan<sup>2</sup>. Naast de chunks die nooit activatie ontvangen is er een groep, die afentoe gedurende 1 tijdstap wordt gebruikt. Deze chunks zijn te herkennen aan de gelijkenis die ze vertonen met een zaagtandfunctie. Meestal worden deze chunks geactiveerd door een gepopte goal, die een eerdere chunk versterkt. Dit opstapelen van activatie kan ervoor zorgen dat de chunk sterk genoeg wordt om na een poosje steeds vaker gebruikt te worden.

In het eerste deel worden meer nieuwe chunks gevormd dan op het eind. Nieuwe ervaringen worden steeds zeldzamer. Tijdens het begin worden er meer nieuwe ervaringen geleerd. Veel van deze ervaringen zijn opgedaan met partners die vanaf de eerste paar tijdstappen al hoge activaties bezaten. Het ACT-R model werkt een soort van conservatisme in de hand. Vaak gebruikte chunks worden steeds vaker gebruikt en minder vaak gebruikte chunks steeds minder. Het

<sup>2</sup>Deze koper heeft inderdaad niet één keer gekozen voor insourcing. Echter hij had nog altijd door het matchingsalgoritme kunnen zijn gedwongen tot insourcing. De koper hoeft namelijk zelf niet in zijn eigen voorkeursvector voor te komen om toch aan zichzelf te worden gekoppeld.

exploitatie versus exploratie probleem is hier van dichtbij te zien. Hoe wordt ervoor gezorgd dat de agent niet vanaf het begin in een status quo terecht komt? Het is vooral de ruis die op de chunkactivatie werkt, die tot het einde toe in staat is om voor afwisseling te zorgen<sup>3</sup>.

### 6.3 Een tweede experiment, met acceptancequota

Onderzocht wordt het effect van het maximaal toelaatbare aantal relaties dat een leverancier kan aangaan. Hebben de economische mechanismen de functies die we van te voren verwachten? Een hoger zogenaamd acceptancequotum voor de leverancier moet grotere schaafeffecten mogelijk maken. Dus de verwachting is een hogere winst voor de kopers bij een hoger quotum.

De parameters voor dit tweede experiment zijn de volgende:

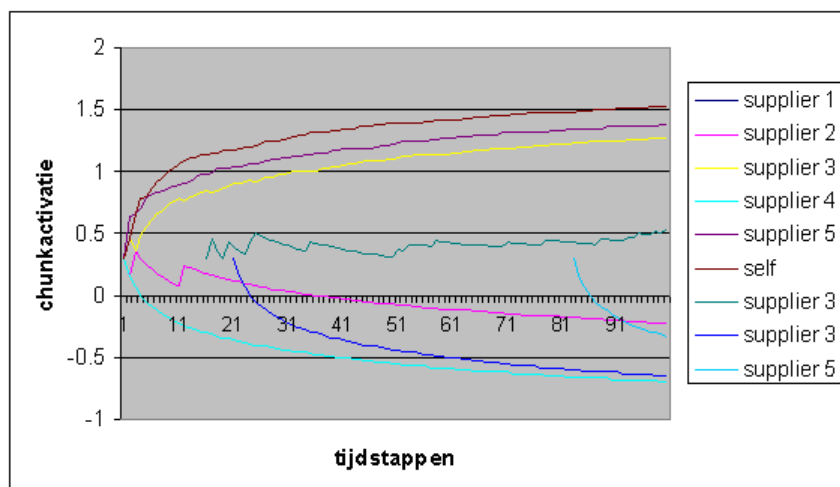
	Parameter	Waarde	Beschrijving
<b>ACTCE</b>	Aantal kopers	5	maar nu ook voor $aq = 1$ kijkende naar slechts 1 differentiatie
	Aantal leveranciers	5	
	Aantal tijdstappen	100	
	Aantal simulaties	100	
	Offerquotum koper	1	
	<b>Acceptancequotum leverancier</b>	<b>{1, 3}</b>	
<b>Differentiaties</b>	<b>0.5</b>		
<b>ACT-R</b>	Base-level learning	0.5	
	Ruisvariantie	0.5	
	Leeftijd van initiële chunks	0	
	Aantal presentaties van initiële chunks	1	
<b>Agentmodel</b>	Afrondingsfactor winst	0.05	
	Zelfrepresentatie koper	true	
	Bovengrens voor winst initiële chunks	0.5	
	Ondergrens voor winst initiële chunks	0.2	
	Eerste satisfiërendrempel voor winst	0.0	
	Tweede satisfiërendrempel voor winst	0.0	
Derde satisfiërendrempel voor winst	0.0		

Tabel 6.2: Parameters voor experiment 2. figuren 6.7 t/m 6.9.

Figuur 6.6 gaf al een blik op het geheugen van een agent, figuur 6.7 doet hetzelfde. Het acceptancequotum is hier echter niet 3 maar 1. Meteen is te zien dat deze agent veel minder chunks heeft en dat er slechts 4 onderling strijden voor activatie. Deze agent heeft geleerd om vooral veel te insourcen, de chunk met de zelfrepresentatie heeft namelijk de hoogste activatie. Bovendien heeft deze agent weinig geleerd. Vanaf het begin heeft hij namelijk weinig relaties gehad met leveranciers. Hij heeft weinig nieuwe ervaringen en weinig vertrouwen opgedaan tijdens de simulatie. Daarom kiest hij vaak voor insourcing.

Dit type agent noemen we Type B en de eerste agent die we zagen, met de vele

<sup>3</sup>Ook wisselende voorkeursvectoren maken het mogelijk voor het matchingsalgoritme om voor nieuwe relaties te zorgen.



Figuur 6.7: Een Type B agent.

chunks in het geheugen, zie figuur 6.6, Type A<sup>4</sup>. Wat blijkt is dat deze Type B agent meer voorkomt bij lagere acceptancequota. Dit wil niet zeggen dat er geen Type A agents zoals in Figuur 6.6 meer voorkomen, maar wel minder vaak. De verklaring voor dit verschijnsel is, dat een lager maximum kopers waarmee een leverancier mag samenwerken, zorgt voor kopers die vaker afgewezen en dus vaker op zichzelf aangewezen worden. Relaties zijn als het ware moeilijker te vormen. Een hoger acceptancequotum zal dus tot minder insourcing leiden.

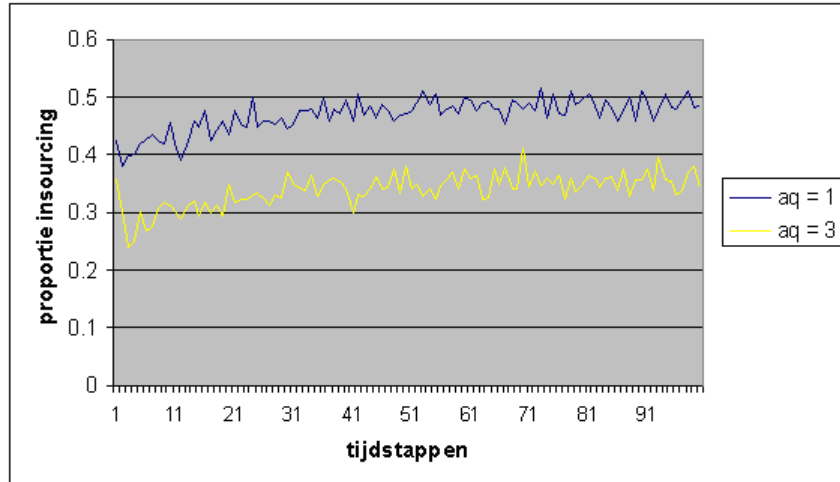
Dit verschijnsel zou onzichtbaar zijn geweest als er niet naar individuele actoren gekeken kon worden. Het idee dat er verschillende typen agents in een populatie aanwezig zijn en dat verschillende omstandigheden de aantallen van elk type verandert is niet afkomstig van TCE. TCE wekt eerder de indruk dat er een gemiddelde actor bestaat die verandert afhankelijk van de situatie.

Kopers gaan inderdaad vaker insourcen bij een lager quotum, zie figuur 6.8. Dit zorgt ervoor dat de winst, anders dan verwacht zoals uit figuur 6.9 blijkt, stijgt. Relaties bereiken ook hier niet een leeftijd waarop schaaffecten zwaarder wegen dan de ongedeelde winst bij insourcing.

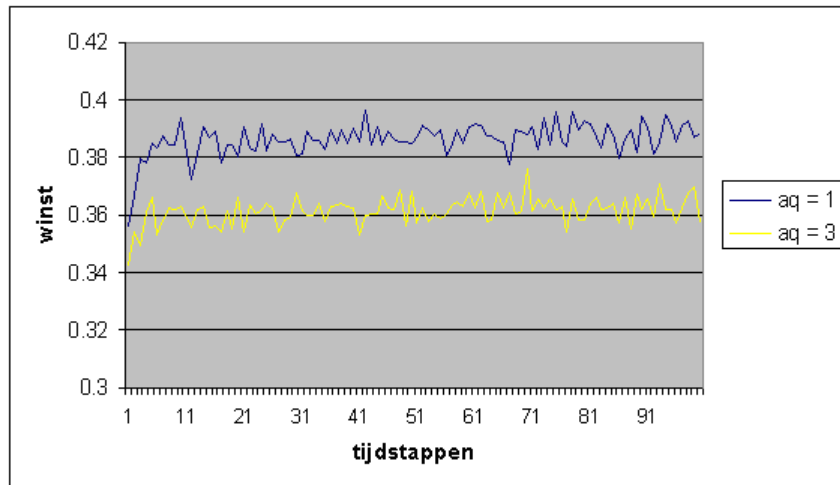
## 6.4 Derde en laatste experiment, met ruis

Is het exploratie versus exploitatie gedrag van de populatie te beïnvloeden door de ruisvariantie te manipuleren? Een plausibele agent moet de juiste balans

<sup>4</sup>Ik wil hier niet beweren dat er een duidelijk gescheiden verdeling van beide typen in een populatie aanwezig zal zijn. Om te zien of er een meer geleidelijke verdeling is, zullen we een methode moeten verzinnen om agents te onderscheiden.

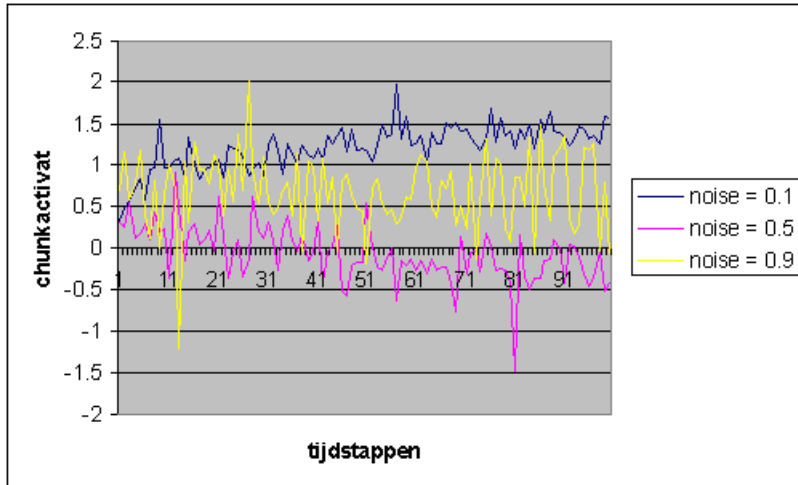


Figuur 6.8: Proportie insourcing.



Figuur 6.9: Gemiddeld behaalde winst.

kunnen vinden tussen deze twee. Om een klein beetje inzicht te krijgen in het effect van de verschillende ruisvarianties, is in figuur 6.10 de verschillende varianties toegepast op een drietal willekeurige chunks.



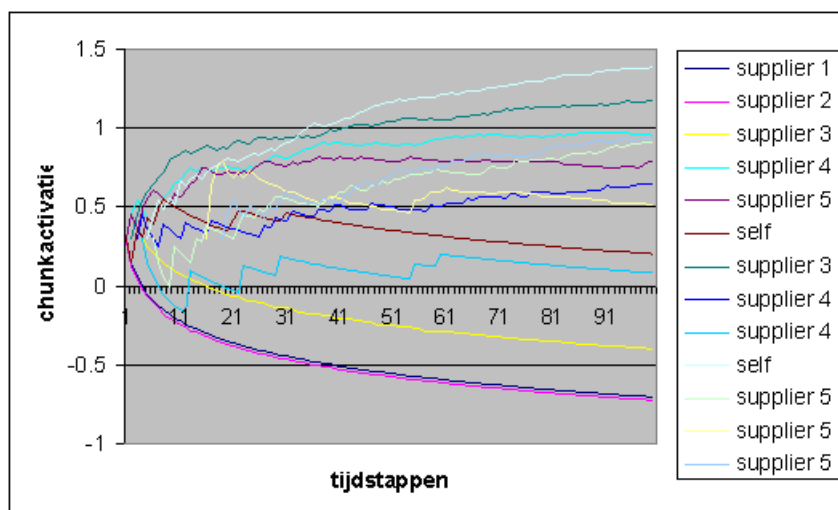
Figuur 6.10: Chunks met verschillende ruisvarianties.

De parameters voor dit experiment zijn de volgende:

	Parameter	Waarde	Beschrijving
<b>ACTCE</b>	Aantal kopers	5	alleen maar voor $aq = 1$
	Aantal leveranciers	5	
	Aantal tijdstappen	100	
	Aantal simulaties	100	
	Offerquotum koper	1	
	<b>Acceptancequotum leverancier</b>	1	
	Differentiaties	0.5	
<b>ACT-R</b>	Base-level learning	0.5	voor verschillende hoeveelheden ruis
	<b>Ruisvariantie</b>	{0.1, 0.5, 0.9}	
	Leeftijd van initiële chunks	0	
	Aantal presentaties van initiële chunks	1	
<b>Agentmodel</b>	Afrondingsfactor winst	0.05	
	Zelfrepresentatie koper	true	
	Bovengrens voor winst initiële chunks	0.5	
	Ondergrens voor winst initiële chunks	0.2	
	Eerste satisfyingdrempel voor winst	0.0	
	Tweede satisfyingdrempel voor winst	0.0	
	Derde satisfyingdrempel voor winst	0.0	

Tabel 6.3: Parameters voor experiment 3. figuren 6.10 t/m 6.15.

Meer ruis laat weer meer kopers van het type A zien, zie figuur 6.11. Ook bij een ruisvariantie van 0.9 zijn Type B agents niet compleet uit de populatie



Figuur 6.11: Een Type A agent bij veel ruis.

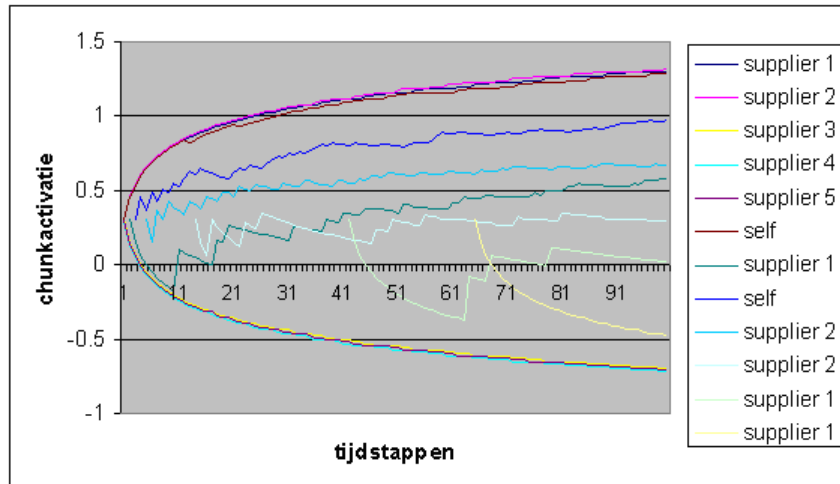
verdwenen. Omgekeerd zijn bij minder ruis de Type B agents in de meerderheid, zie figuur 6.12.

Wat is nu het effect op de populatie? Minder ruis betekent meer insourcing, als we kijken naar figuur 6.13. Foute beslissingen in het begin kunnen minder makkelijk worden overwonnen vanwege het mindere exploration/leereffect. En zoals al eerder te zien was, het Type B heeft een hooggeactiveerde zelfrepresentatie en doet veel aan insourcing. Kijkende naar figuur 6.14 is te zien dat minder ruis langere relaties in de hand werkt. Type B agents veranderen niet zo vaak van partner. Waarschijnlijk zal minder ruis leiden tot meer winst. Figuur 6.15 bevestigt deze verwachting.

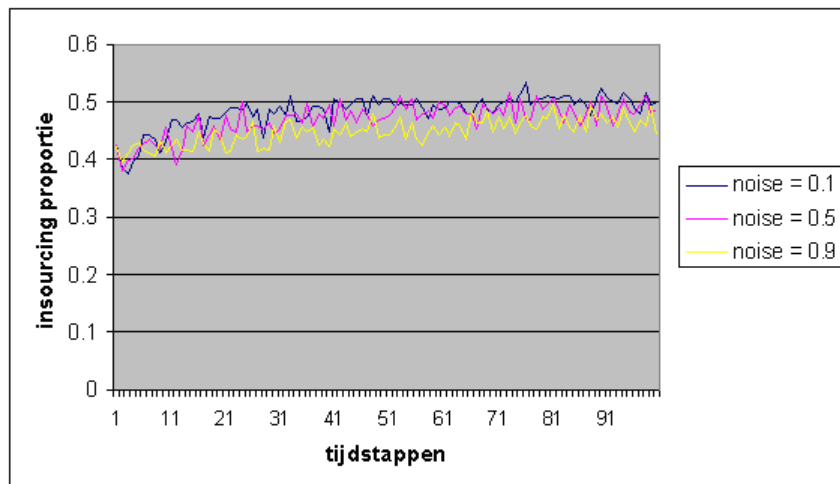
Hiermee zijn de experimenten voor deze scriptie afgesloten. De laatste twee behandelde experimenten waren een bewuste greep uit een hele serie van mogelijke simulaties<sup>5</sup>, die zijn gedraaid tijdens onderzoek naar de werking van het model, met als doel de potentie van ACTCE zichtbaar te maken.

<sup>5</sup>Er zijn simulaties gedraaid waarbij telkens één van de parameters die voorkomen in de tabellen 6.1 t/m 6.4 werd gevarieerd en de overigen constant werden gehouden.

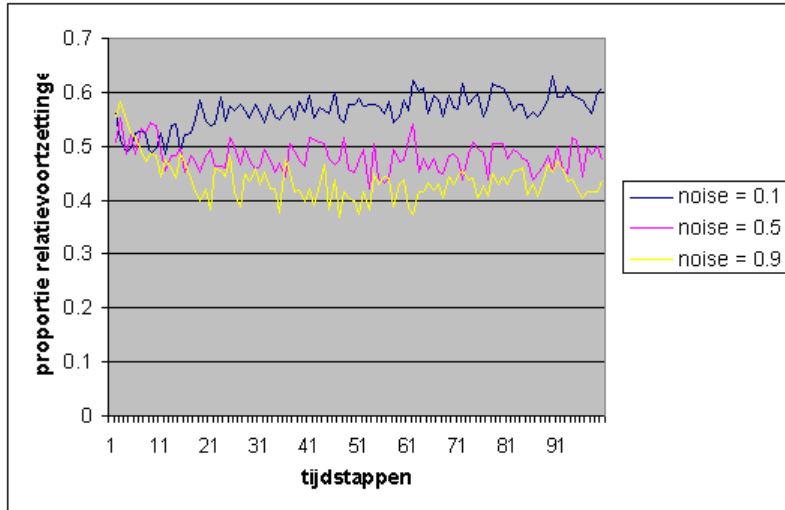




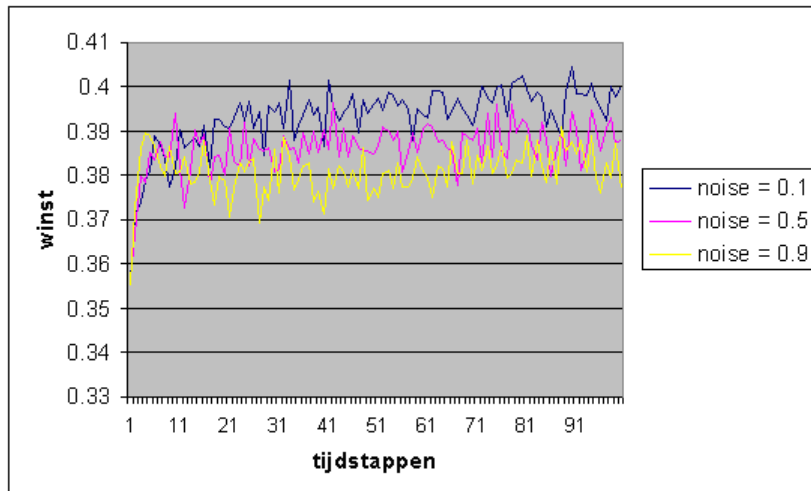
Figuur 6.12: Een Type B agent bij weinig ruis.



Figuur 6.13: Insourcing proporties.



Figuur 6.14: Proportie relatievoortzettingen.



Figuur 6.15: Gemiddeld behaalde winst.

# Hoofdstuk 7

## Conclusies

De onderzoeksvragen die gesteld werden in voorgaande hoofdstukken en de discussiegebieden die hieraan gerelateerd zijn worden hieronder behandeld. Voor alle duidelijkheid worden deze vragen hier herhaald:

- Leiden representaties van transacties tot gedrag dat vertrouwen laat zien? Gaat een bedrijf relaties aan met een ander bedrijf?
- Kan vertrouwen worden geïnterpreteerd uit de gedragingen van een agent in plaats van uit expliciete variabelen? Is aan het gedrag van de agents en de inhoud van de chunks de mate van vertrouwen af te lezen?
- Heeft vertrouwen ervoor gezorgd, dat stabiele vormen van organisatie zich ontwikkelen? Zijn de actoren instaat om om te gaan met onzekerheid?
- Levert vertrouwen meer winst op?
- Moet een markt eerst gevormd worden? Wat is de werking van de marktselectie? Worden er optimale resultaten behaald?

### 7.1 Interpretatie van resultaten

In ons model wordt gedrag gezien in termen van de inhoud en activatie van representaties van transacties aanwezig in het geheugen van een agent. Deze representaties leiden tot keuzes voor bepaalde partners en uiteindelijk tot relaties. Deze relaties kunnen verbroken of voortgezet worden, als gevolg van de eigen keuzes, of door samenspel met de keuzes van andere agents in de simulatie. Het zijn steeds deze drie niveau's (chunkniveau, voorkeursvectorniveau, relatieniveau), die doorlopen moeten worden willen we het gedrag van agents uitleggen. De focus in dit onderzoek ligt zowel op de populatie van agents als op de individuele agents zelf.

Gemiddelde resultaten kunnen tot andere inzichten leiden, omdat de individuele variatie aan het oog wordt onttrokken. Bij het model van Klos was te zien, dat het gebruik van de variabelen  $\alpha$  en  $\tau$  tot duidelijk te begrijpen interactie leidde. Dit werd o.a. duidelijk wanneer de twee variabelen tegen elkaar werden uitgezet. Klos toont aan dat er verschillende attractors aanwezig zijn in de probleemruimte. Dat wil zeggen verschillende punten in de zoekruimte opgespannen door  $\alpha$  en  $\tau$ , waar agents naar toe worden getrokken.

In deze experimenten is een vergelijkbaar verschijnsel waarneembaar. Verschillende agents ontwikkelen zich naast elkaar in de simulatie. Door te kijken naar het declaratief geheugen van agents, is een onderscheid te maken tussen Type A en Type B agents. Bij een Type A agent is relatief veel activiteit in het declaratief geheugen zichtbaar, chunks concurreren daar onderling veel. Een Type B agent vormt echter veel minder nieuwe chunks tijdens een simulatie en laat ook veel minder interactie tussen chunks zien in het geheugen. Verder onderzoek naar de precieze vorming en werking van verschillende typen agents is hier nodig.

Er zijn volgens de resultaten dus verschillende typen agents in een populatie. Veranderingen in de variabelen zorgen niet alleen voor nieuwe typen, maar meer nog voor andere verhoudingen tussen de hoeveelheden bestaande types. Een analyse op het niveau van agents en verdere experimenten is nodig om inzicht in deze processen te krijgen

Eén van de uitgangspunten van deze scriptie was, dat vertrouwen niet expliciet gerepresenteerd mag worden door middel van variabelen, maar impliciet moet komen bovendrijven op het gedrag van de agent zoals zich dat ontwikkelt. De lezer dient vertrouwen te interpreteren op basis van de uitkomsten. Met name kan hierbij gekeken worden naar de grafieken die het aantal relatievoortzettingen weergeven. Meer relatievoortzettingen kunnen immers uitgelegd worden als meer vertrouwen. De grafieken die de inhoud van het declaratief geheugen van een agent weergeven zijn ook geschikt. Met een beetje goede wil kan volgehouden worden, dat hoger geactiveerde representaties van anderen voor een hogere mate van vertrouwen in die ander staat.

Het toevoegen van vertrouwen aan de simulatie heeft er gedeeltelijk voor gezorgd, dat stabiele vormen van organisatie waarneembaar zijn. Het begrip vertrouwen is voortdurend impliciet aanwezig. In de simulaties ontstaan er veel stabiele vormen. Populaties bereiken na een tijdje een stabiel niveau. De individuele agents eindigen vaak in een toestand waarbij een agent steeds wisselt van relatie, maar hierbij zijn huidige partner slechts in de steek laat voor steeds dezelfde subgroep andere partners. Dit in tegenstelling tot wat we zien bij Klos, waar er geen sprake is van de vorming van subgroepen binnen de totale populatie.

Dit gedrag van voortdurend van partner wisselen werkt jammergenoeg het vormen van efficiëntie- en schaaleffecten tegen, zodat deze een minieme rol spelen in het gedrag van de simulatie. Dit komt door het niet of nauwelijks aanwezig zijn van het satisficinggedrag waarmee het ACT-R model is uitgerust. Zoals

te zien was in de tabellen met simulatieparameters staan de drie drempelwaarden waarop een ervaring kan worden geselecteerd op nul. Als dit niet wordt gedaan en chunks alleen meedingen om geselecteerd te worden als hun profit slot wat betreft inhoud hoger was dan de drempel, dan kiezen de agents nog vaker voor insourcing dan anders. Door het grote initiële verschil in opbrengst tussen de keuze voor insourcing en outsourcing, in het beginstadium van een relatie scheelt dit namelijk een factor twee in het voordeel van insourcing, kunnen er geen juiste drempels gevonden worden om de twee keuzes tegen elkaar af te wegen. De keuze wordt dan altijd beslecht in het voordeel van insourcing. Om voldoende outsourcing mogelijk te maken zijn de drempels verlaagd.

## 7.2 Initiëel aanpassingsproces

De resultaten laten zien dat elke simulatie een starteffect heeft. Het kost tijd om de markt te leren kennen en een goede partner te vinden. En net zo belangrijk, het kost tijd voor ervaringsefficiëntie om een rol te spelen. De resultaten tonen het belang aan van het maken van een markt. Tijdens dit proces wordt bepaald welke rol een agent op zich neemt (welk type hij wordt). In een beginsituatie moet een markt zich nog vormen en de agents passen zich voortdurend aan. Dit leidt echter niet altijd tot samenwerking. Agents leren slechts zelden dat langdurige relaties tot de meeste winst leiden.

Wanneer dit aanpassingsproces tot rust komt is de kans dat agents optimale relaties hebben gevormd echter praktisch nul. Anders dan TCE aanneemt is er veel meer coördinatie nodig om tot optimale resultaten te komen, dan de individuele agents bezitten. Op de vraag of agents met beperkte rationaliteit optimale oplossingen kunnen vinden, moet negatief geantwoord worden. Telkens zijn namelijk groepen autonome agents zichtbaar, die in de meeste gevallen suboptimale oplossingen vinden in plaats van optimale oplossingen.

## 7.3 Verder onderzoek

Het eerste waarna gekeken moet worden bij verder onderzoek, is het niet of nauwelijks aanwezig zijn van het satisficinggedrag waarmee het ACT-R model is uitgerust. De drempelwaarden moeten een functie krijgen. Dit kan door ofwel verschillende procedures met verschillende drempels te maken voor de keuze van insourcing en outsourcing. Of door de feedback vanuit het economisch model dichter naar elkaar toe te brengen, dit betekent minder verschil tussen insourcing en outsourcing opbrengsten. Zodat beide keuzes met dezelfde drempels, voor wat een goede winst is en welke een mindere, kunnen worden genomen.

De voorkeursvectoren worden nu nog door een matchingsalgoritme verwerkt tot een serie koppelingen tussen kopers en leveranciers. In de toekomst kan dit algoritme worden vervangen door actieve verkenning van en zoeken op de markt door de agents zelf.

De voorzichtige interpretatie die aan de resultaten is gegeven wat betreft de mate van aanwezig vertrouwen, is de volgende. Een hogere mate van insourcing en een laag aantal relatievoortzettingen getuigt van weinig vertrouwen. Of, hoog geactiveerde chunks representeren betrouwbare agents. Andere methoden om vertrouwen meetbaar te maken zijn denkbaar. Uit de reeksen zou naast het aantal relatievoortzettingen per populatie ook nog de volgende informatie kunnen worden geïnterpreteerd als maten van vertrouwen:

- Het totaal aantal relaties met een partner (dit is vertrouwen in een bepaalde actor). Het aantal transacties met zichzelf zou een maat van globaal wantrouwen kunnen zijn.
- Het aantal reeksen van opeenvolgende relaties met dezelfde partner kan een maat zijn voor vertrouwen in die bepaalde actor. Het totale aantal series onverschillig met welke partner, is dan een maat voor globaal vertrouwen.
- Het aantal relatievoortzettingen, per actor levert een maat van vertrouwen. Het totale aantal relatieverbrekingen is een algemene maat.

Deze tellingen kunnen losgelaten worden op een enkele actor, maar ze zouden ook nog eens gesommeerd kunnen worden over alle actoren, of over een deel daarvan, bijvoorbeeld alle kopers.

Ervaringen in dit model gaan over behaalde winst met een bepaalde partner. Elke actie levert voor een koper winst op en nooit verlies. Bovendien neemt de actor representaties met een te lage winst niet mee in zijn overwegingen. Het maakt hem niet uit dat hij een partner kiest waarmee hij één keer veel winst scoorde terwijl hij de rest van de tijd hele lage winsten met hem heeft gehad. Door de agent ook slechte ervaringen te laten vormen, door negatieve feedback vanuit het economische deel van de simulatie en door de agent ook de mogelijkheid te geven partners nadrukkelijk niet te kiezen vanwege slechte ervaringen, zal er een veel realistischer beeld ontstaan van het gedrag van de agents. Wil een agent ook om kunnen gaan met negatieve ervaringen, dan moeten het agentmodel uitgebreid worden met productieregels die ook letten op agents waarmee lage winsten zijn behaald. Een uitbreiding op de huidige productieregels zou kunnen zijn, een agent niet als partner te kiezen wanneer deze in het verleden betrokken was bij een transactie met erg weinig behaalde winst.

Tenslotte zou het interessant zijn om te zien hoe een nieuwe agent zich gaat gedragen, wanneer hij in een al voor langere tijd bestaande populatie wordt geïntroduceerd. Gedacht kan worden aan verschillend geïntialiseerde agents. Bijvoorbeeld een opportunistische agent in een verder loyale populatie, of omgekeerd. Handig zou tevens zijn de mogelijkheid om niet alleen bij de initialisatie het geheugen van een agent te manipuleren, maar dit op elk gewenst moment in de simulatie te kunnen doen. De wisselwerking die hieruit zal ontstaan maakt het interessant om te achterhalen wat de invloed van de initialisatie is op het eindresultaat.

# Bibliografie

- Aha, D., Kibler, D., and Albert, M.K. (1991). Instance based learning algorithms. *Machine Learning* volume 6, pages 37-66, 1991.
- Anderson, J.R. (1990). *The Adaptive Character of Thought*. Redwood City CA: Benjamin/Cummings
- Anderson, J. R. (1993). *Rules of the mind*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Anderson, J.R. & Lebiere, C. (1998). *The atomic components of thought*. Mahwah, NJ: Lawrence Erlbaum Associates.
- Axelrod, R. (1980a). Effective Choice in the Prisoner's Dilemma. *J, Confl. Resol.*, 24: 3-25
- Axelrod, R. (1980b). More Effective Choice in the Prisoner's Dilemma. *J. Confl. Resol.*, 24: 379-403.
- Axelrod, R. (1984). *The Evolution of Cooperation*. New York: Basic Books.
- Axelrod, R. & Hamilton, W.D. (1981). The evolution of cooperation. *Science* 211: (4489) 1390-1396
- Baron, R. A., Byrne, D., & Suls, J. (1989). *Exploring social psychology* (3rd Ed.). Boston: Allyn and Bacon.
- Batson, C.D., O'Quin, K., Fultz, J., & Vanderplas, M. (1983). Influence of self-reported distress and empathy on egoistic versus altruistic motivation to help. *Journal of Personality and Social Psychology*, 45 (3), 706-718.
- Batson, D., Fultz, J., & Schoenrode, P. (1987). Distress and empathy: Two qualitatively distinct vicarious emotions with different motivational consequences. *Journal of Personality*, 155, 19-39.
- Birk, A. (1999). Learning to Trust. *Trust in Cyber-societies 2000*: 133-144

- Boon, S.D. & Holmes, J.G. (1991). The dynamics of interpersonal trust: resolving uncertainty in the face of risk. Hinde, R.A. & Groebel, J. (eds.): Cooperation and Prosocial Behaviour, Cambridge University Press, 190-211.
- Broek, van den, H. (2001). On agent cooperation: the relevance of cognitive plausibility for multi-agent simulation models of organization. University of Groningen: SOM-Thesis.
- Castelfranchi, C. & Falcone, R. (1999). The Dynamics of Trust: from Beliefs to Action. Autonomous Agents '99 Workshop on "Deception, Fraud and Trust in Agent Societies", Seattle.
- Castelfranchi C., & Falcone R., (2000). Trust is much more than subjective probability: Mental components and sources of trust, 32nd Hawaii International Conference on System Sciences - Mini-Track on Software Agents, Maui, Hawaii, 5-8 January 2000.
- Cialdini, R. B., Schaller, M., Houlihan, D., Arps, K., Fultz, J., & Beaman A. L. (1987). Empathy-based helping: Is it selflessly or selfishly motivated? Journal of Personality and Social Psychology, 52, 749-758.
- Coase, R.H. (1937). The Nature of the Firm. *Economica* NS, 4 (16), p.386-405.
- Conte, R., Hegselmann, R. & Terna, P. (Eds.) (1997). Simulating Social Phenomena Berlin: Springer.
- Darley, J. M., & Batson, C. D. (1973). From Jerusalem to Jericho: A study of situational and dispositional variables in helping behavior. Journal of Personality and Social Psychology, 27, 100-108.
- Denes-Raj, V., & Epstein, S. (1994). Conflict between intuitive and rational processing: When people behave against their better judgment. Journal of Personality and Social Psychology, 66 (5), 819-829.
- Dennett, D. C. (1981). Brainstorms: Philosophical Essays on Mind and Psychology. Cambridge: Bradford Books / MIT Press.
- Dennett, D. C. (1987). The intentional stance. Cambridge (Mass): The MIT Press.
- Deutsch, M. (1958). Trust and Suspicion. Journal of Conflict Resolution, volume 2 (4) 265-79.
- Deutsch, M. (1973). The Resolution of Conflict. New Haven: Yale University Press.
- Eisenberg, N., & Miller, P. A. (1987). The relation of empathy to prosocial and Related behaviors. Psychological Bulletin, 101, 91-119.



- Eisenberg, N., Fabes, R. H., Miller, P. A., Fultz, J., Shell, R., Mathy, R. M., & Reno, R. (1989). Relation of sympathy and personal distress to prosocial behavior: A multimethod study. *Journal of Personality and Social Psychology*, 57, 55-66.
- Epstein J. and R. Axtell (1996). *Growing Artificial Societies: Social Sciences from the Bottom Up*. Brookings, MA: MIT Press.
- Friedman, M. (1953). *The Economics of Positive Methodology*. In *Essays on Positive Economics*. Chicago: University of Chicago Press).
- Gambetta, D., (1988). Can we Trust Trust?, in Gambetta ed. *Trust: The Making and Breaking of Cooperative Relations*. Oxford: Basil Blackwell.
- Gazendam, H.W.M. & Jorna, R.J. (1998). *Theories about architecture and performance of multi-agent systems*. University of Groningen: SOM Research Report 98 A02.
- Gazendam, L. & Jorna, R.J. (2002). *Transaction Cost Economics and Plausible Actors: A Cognitive Reappraisal*. Internal report: University of Groningen.
- Gray, P. (1991). *Psychology* (second edition). Worth publishers, inc.
- Gulati, R. (1995). Does familiarity breed trust? The implications of repeated ties for contractual choice in alliances. *Academy of Management Journal* 38(1), 85-112.
- Hardin, G. (1968). *The Tragedy of the Commons*. *Science*, Vol. 162, pp. 1243-1248.
- Helmhout, J.M. (2001). *Make or Buy? Een beslissing gebaseerd op de transactiekostentheorie*. Master Thesis Fac. Bedrijfskunde, Rijksuniversiteit Groningen.
- Hewitt, C.E. (1977). Viewing control structures as patterns of passing messages. *Journal of Artificial Intelligence*. Vol. 8, No. 3, June 1977, pp. 323-364.
- Hill, C.W.L. (1990). Cooperation, Opportunism, and the invisible hand: Implications for Transaction Cost Theory. *Academy of Management Review* 15 (3), 500-513.
- Holland, John H. (1992). Genetic Algorithms, *Scientific American*, June, 66-72.
- Jager, W. (2000). *Modelling Consumer Behaviour*. Universal Press. Proefschrift Rijksuniversiteit Groningen.

- Jorna, R.J. (2001). De 'zwarte doos' in de bedrijfskunde: cognitie in actie. *Nederlands Tijdschrift voor de Psychologie*, 55, 6, pp 287-295. (The black box in management science: cognition in action)
- Klos, T. B. (1999). Governance and Matching. Paper for the Society for Computational Economics fifth International Conference on Computing in Economics and Finance, Boston College, Chestnut Hill, MA, June 24–26.
- Klos, Tomas B. (1999). Decentralized Interaction and Co-Adaptation in the Repeated Prisoner's Dilemma. *Computational and Mathematical Organization Theory*.
- Klos, T.B (2000). Agent-based Computational Transaction Cost Economics. Ridderkerk, Labyrinth Publication.
- Klos, T.B. & Bart Nooteboom (1998). Adaptive Governance: The Role of Loyalty. Santa Fe Institute Economics Working Paper 98-06-048E.
- Knapik, M. & Johnson, J. (1998). Developing Intelligent Agents for Distributed Systems. Exploring Architecture, Technologies, and Applications. McGraw-Hill
- Lebiere, C. & Wallach, D. (1998). Implicit does not imply procedural: A declarative theory of sequence learning. Paper presented at the 41st Conference of the German Psychological Association, Dresden, Germany.
- Lebiere, C. , Wallach, D. & West, R.L. (2000). A Memory-based Account of the Prisoner's Dilemma and Other 2x2 Games. *Proceedings of International Conference on Cognitive Modeling 2000*, pp. 185-193.
- Lebiere, C. & West, R.L. (1999). A Dynamic ACT-R Model of Simple Games. In *proceedings of the twenty first annual conference of the cognitive science society*, pp.296-301. Mahwah, NJ:Erlbaum.
- Maes, P. (1995). Modelling adaptive autonomous agents. In C. Langton, editor, *Artificial Life: An Overview*, pages 135-162, Cambridge, MA, MIT Press.
- Marsh, S.P. (1994). Formalising Trust as a Computational Concept. Ph.D. Thesis, Department of Computing Science and Mathematics, University of Stirling.
- Masuch, M. & Warglien, M. (eds.) (1992). *Artificial Intelligence in Organization and Management Theory: Models of Distributed Activity*. Amsterdam: North-Holland.
- McGovern, L., Ditzian, J., and Taylor, S. (1975). The effect of one positive reinforcement on helping with cost. *Bulletin of the Psychonomic Society*, 5, 421-423.

- Miller, J.H. (1996). The coevolution of automata in the repeated prisoner's dilemma. *Journal of Economic Behavior and Organization* 29(1), 87112. Originally published as (Miller 1989).
- Nash, J. (1950). Equilibrium points in N-person games. *Proceedings of the National Academy of Sciences*, 36, 48-49.
- Newell, A. (1990). *A Gentle Introduction to Soar*.
- Nooteboom, B. (1994). Cost, Quality and Learning Based Governance of Transactions: Western, Japanes and a Third Way, paper EARIE conference, Crete, september, and EMOT conference, Como october.
- Nooteboom, B. (1994). *Management van partnerships*, 2nd edition. Schoonhoven: Academic service.
- Nooteboom, B. (1995). *Trust, Opportunism and Governance: A process and Control Model*. *Bedrijfskunde Groningen*
- Nooteboom, B. (1999b). *Inter-Firm Alliances*, Routledge, London
- Nooteboom, B., H. Berger, N.G. Noorderhaven (1995). Sources, Measurement and Effect of Trust in the Governance of Buyer-supplier Relations, Research Report 95B39,
- Numan, J.H. (1998). *Knowledge based Systems and Companions*. Capelle aan de IJssel Labyrinth Publications
- Robbins, S. P. (2002). *Essentials of Organizational Behavior* (7th edition). Englewood Cliffs, NJ: Prentice Hall.
- Russel, S.J. & Norvig, P. (1995). *Artificial Intelligence. A Modern Approach*. Prentice-Hall International, Inc.
- Sargent, T.J. (1993). *Bounded Rationality in Macroeconomics*. Oxford: Clarendon Press.
- Schelling, T. (1978). *Micromotives and Macrobehavior*. (New York: W.W. Norton).
- P.L. Schindler en C.C. Thomas (1993). The Structure of Interpersonal Trust in the Workplace, *Psychological Reports*, October 1993, pp. 563-73
- Simon, H.A. (1956). Rational choice and the structure of the environment. *Psychological Review*, 63, 129-138.
- Simon, H.A. (1958). Rational choice and the structure of the environment. In *models of Bounded Rationality*, volume 2. MIT Press, Cambridge (Mass).
- Simon, H.A. (1972). Theories of bounded rationality. In C.B. Radner & R.Radner (eds.), *Decision and Organization* (pp. 161-176) Amsterdam: North-Holland.

- Kulkarni, D. & Simon, H.A. (1990). The processes of scientific Discovery: The strategy of experimentation. *Cognitive Science*, 12, 139-176.
- Steels, L. (ed)(1995). *The Biology and Technology of Intelligent Autonomous Agents*. NATO ASI Series: Series F, Computer and Systems Sciences, vol. 144, Berlin: Springer-Verlag.
- Steels, L. and Brooks, R. (eds) (1995). *The 'Artificial Life' route to 'Artificial Intelligence'*. Building Embodied, Situated Agents. New Haven: Lawrence Erlbaum Associates.
- Taatgen, N. A. (1999). *Learning without limits: From problem solving toward a unified theory of learning*. Ph.D. Thesis, Department of Psychology, University of Groningen, The Netherlands.
- Tesfatsion, L. (1997). *Syllabus of Readings for Artificial Life and Agent-Based Economics* [<http://www.econ.iastate.edu/tesfatsi/sylalife.htm>]
- Todd, P.M. & Gigerenzer, G. (1999). *Simple heuristics that make us smart*. Oxford University Press.
- Todd, P.M. & Gigerenzer, G. (2000). *Prcis of Simple heuristics that make us smart*. *Behavioral and Brain Sciences* (2000) 23, 727-780.
- Weiss, G. (Eds.) (1999). *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. The MIT Press.
- Williamson, O.E. (1975). *Markets and Hierarchies: Analysis and Antitrust Implications*. The Free Press. New York.
- Williamson, O.E. (1979). *Transaction cost economics: The governance of contractual relations*. *Journal of Law and Economics* 22 (2), 233-261.
- Williamson, O.E. (1981a). *The economics of organization: The transaction cost approach*. *American Journal of Sociology* 87 (3), 548-577.
- Williamson, O.E. (1985). *The Economic Institutions of Capitalism: Firms, Markets, Relational Contracting*. New York: The Free Press.
- Williamson, O.E. & Winter, S.G. (1993). *The Nature of the Firm. Origins, Evolution, and Development*. Oxford: Oxford University Press
- Wispe, L. (1986). *The distinction between sympathy and empathy: To call forth a concept, a word is needed*. *Journal of Personality and Social Psychology*, 50, 314- 321.

## Bijlage A

# Het ACT-R model

Hieronder een ACT-R model van een agent genitialiseerd met vijf representaties van partners, waar hij uit mag kiezen, in zijn declaratief geheugen.

```
.....  
;;;  
;;; The ACT-R model  
;;;  
  
(clearall)  
  
.....  
;;;  
;;; Define the chunk types for the model  
;;;  
  
(chunk-type episode name eProfit profit first second third )  
  
.....  
;;;  
;;; Initialize the working memory for the model  
;;;  
  
(add-dm  
(Agent_1  
  isa    episode  
  name  1  
  profit (!eval! (reasonableRandom)))  
(Agent_2  
  isa    episode  
  name  2  
  profit (!eval! (reasonableRandom)))
```

```

(Agent_3
  isa    episode
  name   3
  profit (!eval! (reasonableRandom)))

(Agent_4
  isa    episode
  name   4
  profit (!eval! (reasonableRandom)))

(Agent_5
  isa    episode
  name   5
  profit (!eval! (reasonableRandom)))

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;
;;; the main goal of the model
;;;

(goal isa episode)
;;; focus on the main goal
(goal-focus goal)

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;
;;; PRODUCTIONS
;;;

(p pickFirst
  =goal>
    ISA    episode
    first  nil
==>
  +retrieval>
    ISA    episode
    name   =name
  > profit 0.5
  =goal>
    first  =name)

```

```

(p pickSecond
 =goal>
   ISA    episode
   first  =first
   second nil
==>
 +retrieval>
   ISA    episode
   name   =name
 - name   =first
 > profit 0.25
 =goal>
   second =name)

(p pickThird
 =goal>
   ISA    episode
   first  =first
   second =second
   third  nil
==>
 +retrieval>
   ISA    episode
   name   =name
 - name   =first
 - name   =second
 > profit 0.125
 =goal>
   third  =name)

```





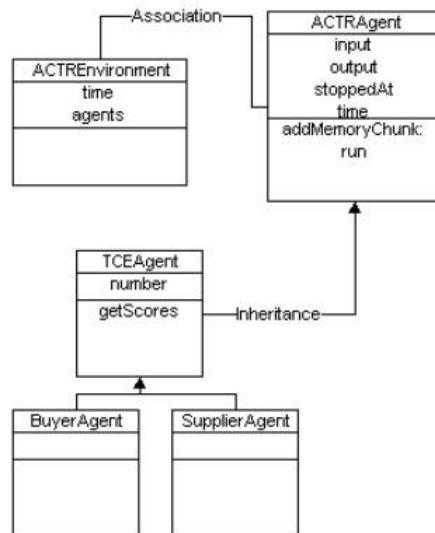
```
(p pickThirdFail
  =goal>
    ISA    episode
    first  =first
    second =second
    third  nil
==>
  =goal>
    third  'retrievalError')
```

---

## Bijlage B

# ACT-R, Object Georiënteerd

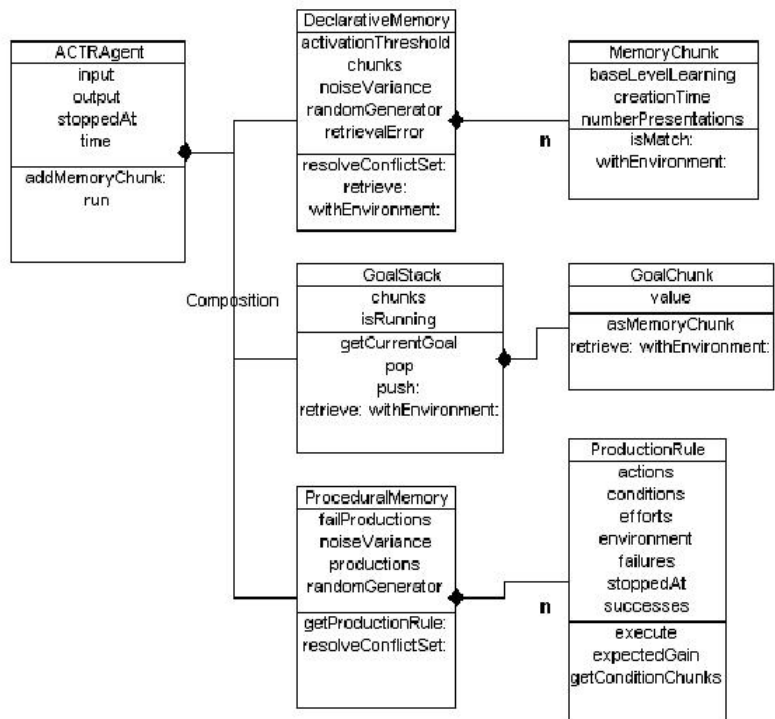
Zie figuur B.1, bovenaan de Hiërarchie staat het ACT-R model (ACTRAgent); Onze ACT-R omgeving (ACTREnvironment) kan meerdere van deze agents bevatten. Omdat de simulatie meer verlangt dan dat tot de ACT-R theorie behoort, hebben we subclasses gecreëerd waarin meer economisch gerelateerd gedrag is gedefiniëerd (TCEAgent, BuyerAgent, SupplierAgent).



Figuur B.1: Objectmodel met vijf classes hoog in de hiërarchie.

De agent verder onder de loep nemend in figuur B.2, zien we dat hij bestaat uit

drie delen. Een declaratief geheugen (DeclarativeMemory), welke een willekeurig aantal chunks (MemoryChunk) bevat. Een GoalStack die in de vorige versie 4.0 nog meerdere goal kon bevatten. Wij houden op dit punt versie 5.0 aan, dat betekent dat er altijd maar sprake is van n goal, de huidige<sup>1</sup>. Als laatste is er ook nog het procedurele geheugen (ProceduralMemory). Hierin kunnen zich meerdere productieregels (ProductionRule) bevinden.

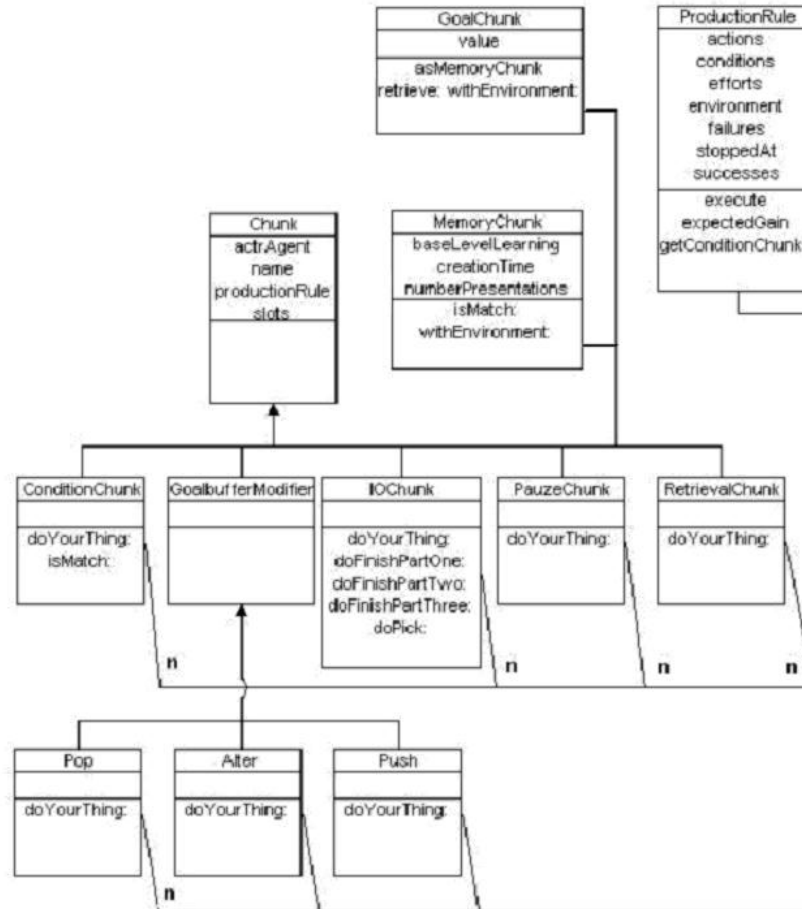


Figuur B.2: De ACT-R architectuur als een objectmodel.

De chunk maken we een superclass van een heleboel subclasses. De meeste van de subclasses hebben een eigen functie binnen een productieregel. De productieregel is niet anders dan een geordende verzameling chunks die gezamenlijk een bepaalde handeling verrichten, waarbij de chunks de bouwstenen vormen. Normaal gesproken wordt een regel niet opgedeeld in chunks, maar het ligt voor de hand om dit vanuit een object georiënteerd oogpunt wel te doen. Figuur B.3 laat de verschillende chunks zien. Ook de twee chunks die ACT-R erkent, chunks in het declaratief geheugen (MemoryChunk) en chunks op de

<sup>1</sup>De reden dat we toch gebruik maken van een zogenaamde class GoalStack, is om het wisselen tussen de verschillende ACT-R versies te vergemakkelijken, mochten we dit willen.

goalstack (GoalChunk), erven de eigenschappen van de class Chunk.



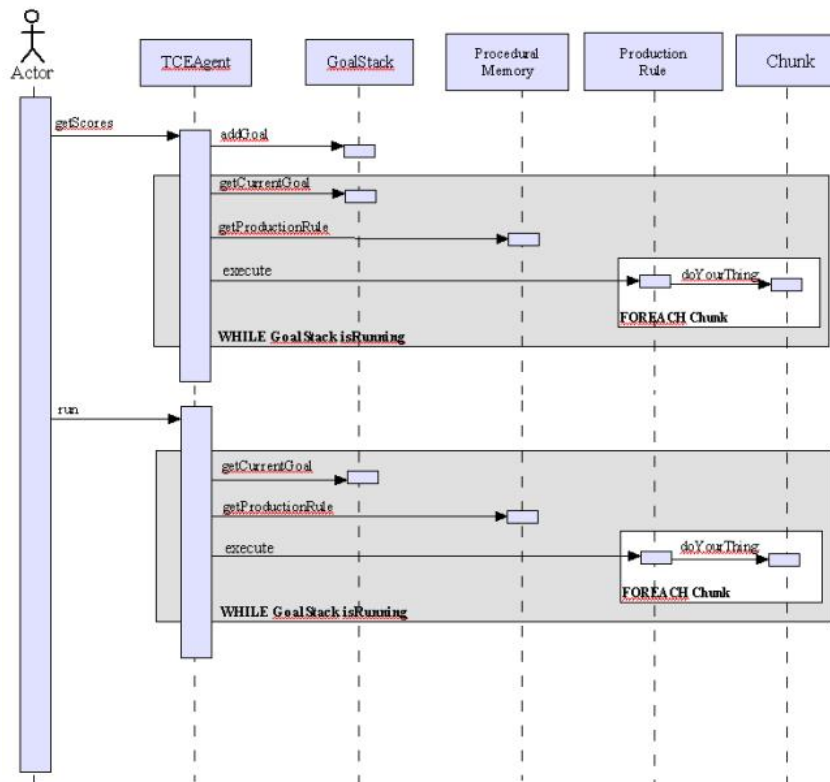
Figuur B.3: Objectmodel met verschillende chunks.

In het kort een opsomming van het gedrag van de verschillende chunks die in een productieregel kunnen voorkomen.

- De ConditionChunk vormt een beschrijving van de slots waar de goal aan moet voldoen wil de regel meedingen in de strijd om te mogen vuren.
- De IOChunk regelt de interface (communicatie) met de rest van de simulatie waar dat nodig is. In deze class moeten dan ook waarschijnlijk nieuwe interfaces worden gedefinieerd telkens als er buiten ACT-R iets verandert wat de uitwisseling van informatie beïnvloedt.

- De PauzeChunk maakt het mogelijk om een agent midden in zijn executie te stoppen. Wanneer de executie weer verder kan, gebeurt dit vanaf dezelfde positie.
- De RetrievalChunk kijkt of er een chunk in het declaratief geheugen is die voldoet aan de opgegeven specificaties.
- De laatste drie Pop, Alter, Push zijn die de goal veranderen. Ze respectievelijk verwijderen, veranderen of maken een goal.

In figuur B.4 zien we een scenario view van het beslissingsproces van een agent gedurende een complete tijdstep. Samen met het bovenstaande objectmodel ontstaat dan een redelijk beeld van de werking van ACT-R in een OOP taal.



Figuur B.4: Een scenarioview van het beslissingsproces van een actor.

Het uitvoeren van de desbetreffende productieregels vindt plaats in de FOREACH lussen. De keuze en opvolging van chunks binnen een productieregel bepaalt hier het handelen van de agents.

## Bijlage C

# ACT-R, de theorie

ACT-R is een geünificeerde theorie van cognitie (Anderson en Lebiere, 1998)<sup>1</sup>. ACT staat voor "The Atomic Components of Thought" of volgens sommigen ook wel voor "Adaptive Control of Thought". De 'R' staat voor 'rational analysis', elk onderdeel van de architectuur, hoe beperkt ook in computationeel vermogens, streeft voortdurend naar optimalisatie in relatie tot de omgeving. Het doel van de theorie is om een architectuur voor cognitie te ontwikkelen welke in detail een verzameling cognitieve taken kan uitvoeren. De architectuur heeft de vorm van een computersimulatie die dezelfde taken kan uitvoeren en daarvan kan leren als proefpersonen. Het doel is te begrijpen hoe wij kennis vergaren en organiseren en intelligent gedrag produceren.

ACT-R is een voorbeeld van een hybride theorie. Kennis wordt namelijk niet alleen symbolisch als regels en feiten gerepresenteerd, maar kent ook een op neuronen gebaseerd activatie-proces dat bepaalt welke regels en feiten worden gebruikt in welke situatie.

ACT-R<sup>2</sup> bestaat uit:

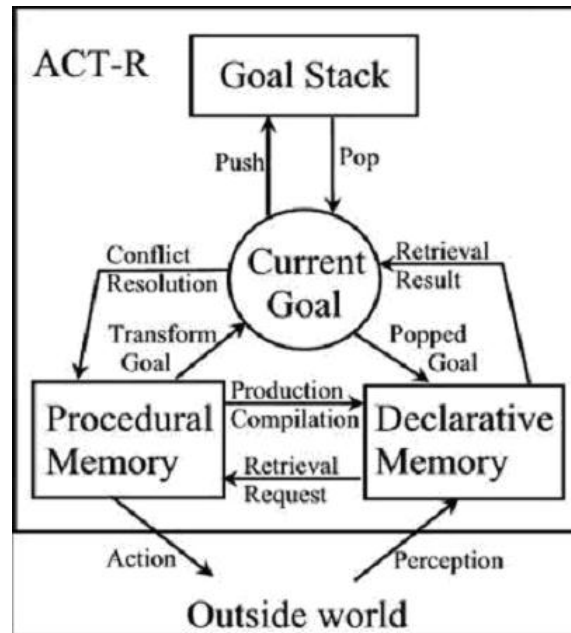
- een goalstack, met daarop goals geformuleerd in de vorm van chunks.
- Een declaratief geheugen gevuld met chunks.
- Een procedureel geheugen gevuld met productieregels.

Aan de hand van een rekentaak, waarbij het doortellen op de vingers bij een-

---

<sup>1</sup>ACT-R is één van de succesvolste theorieën over cognitief modelleren. Zie ook Newell (1990) voor een introductie in Soar, een andere belangrijke architectuur voor cognitie.

<sup>2</sup>Inmiddels wordt er tijdens dit schrijven gewerkt aan een nieuwe versie van ACT-R. Afgezien een paar punten zijn versies 4.0 en 5.0 gelijk. Deze punten van verschil zijn, het verdwijnen van de goalstack in versie 5.0. In ACT-R 5.0 is het nu mogelijk een retrieval request aan de conditiekant van de productieregel te definiëren.



Figuur C.1: Een overzicht van de ACT-R architectuur.

voudige optelsommen wordt gemodelleerd<sup>3</sup>, zullen we deze onderdelen behandelen.

## C.1 Goalstack

ACT-R is een theorie over probleemoplossing, die ervan uitgaat dat menselijke cognitie (althans hogere, symbolische, bewuste cognitie) uitgaat van een op te lossen probleem dat als doel (goal) wordt geformuleerd. Dit kan door bijvoorbeeld subdoelen te formuleren, die gemakkelijker op te lossen zijn dan het hoofddoel. Er is altijd sprake van een huidige goal, zij het hoofd- of subgoal, die opgelost dient te worden.

Een goal kan er uitzien als in tabel C.1. Zo een goal wordt gebruikt om een bijpassende productieregel uit het procedureel geheugen te selecteren.

<sup>3</sup>Het voorbeeld wat we hier gebruiken komt uit de ACT-R documentatie. Zie de ACT-R website, <http://act-r.psy.cmu.edu/>

Tabel C.1: Een voorbeeld van een goal.

ACT-R		Beschrijving
<code>first-goal:</code>		De naam van de goal.
<code>isa</code>	<code>count-from</code>	Het type van de goal.
<code>start</code>	2	En een aantal naar eigen inzicht in te richten
<code>end</code>	5	slots gevuld met kennis of associaties.
<code>step</code>	<code>start</code>	In dit geval drie stuks.

## C.2 Procedureel geheugen

De productieregel (ALS-DAN-regel) bestaat uit twee delen. De conditie van een productieregel (het ALS gedeelte) bestaat uit een beschrijving van een goal waaraan de huidige goal moet voldoen, wil de regel vuren. Ook kan er hier een chunk uit het geheugen worden opgehaald, of althans een poging hiertoe worden gedaan. Chunks zijn gegroepeerde stukken kennis opgeslagen in het declaratief geheugen dat in de volgende paragraaf wordt behandeld. Aan de actiekant (het DAN gedeelte) worden nieuwe chunks gevormd in het declaratief geheugen, chunks uit het declaratief geheugen opgehaald, of de huidige goal aangepast. Productieregels opereren op een probleem en passen dit meestal stap voor stap aan, totdat het is opgelost.

ACT-R heeft zijn eigen aparte notatiewijze, dit komt omdat het geheel is geschreven in de programmeertaal LISP. Producties voeren operaties uit op goal-chunks en declaratieve chunks.

Wanneer geen enkele productieregel mag vuren, omdat niet aan de conditie is voldaan, wordt er gekeken of er een fail-regel van toepassing is. Deze fail-regels moeten we net als de 'normale' regels ook zelf implementeren.

Elke productie kent een utiliteitswaarde, welke een maat geeft voor de te verwachten bijdrage in het bereiken van een goal. Dit is wat Anderson (1990) bedoelt met 'rational analysis'. De utiliteitswaarden streven voortdurend naar optimalisatie in relatie tot de omgeving, ze proberen een zo efficiënt mogelijke verdeling te vinden. De huidige goal selecteert een set producties die op deze goal van toepassing zijn en de productieregel uit deze set met de hoogste utiliteit is uiteindelijk de enige die vuurt. Drie factoren worden gebruikt om een utiliteitswaarde te berekenen.

$$Utility = U = PG - C \quad (C.1)$$

*P*: De waarschijnlijkheid dat de productie succesvol de goal zal behalen. *C*: De verwachte kosten van het bereiken van de goal. De kosten representeren het aantal seconden vanaf het moment dat de productie wordt geselecteerd tot aan het bereiken van de goal. *G*: De waarde van de goal, uitgedrukt ook in seconden. Deze waarde wordt niet door ACT-R geleerd, maar door de modelleerder met de hand gezet.



Tabel C.2: Een voorbeeld van een productieregel.

ACT-R	Beschrijving
<pre>(P increment =goal&gt;   isa      count-from   start    =num1 - end      5   step     start =retrieval&gt;   isa      count-order   first    =num1 - second   =num2 ==&gt; =goal&gt;   start    =num2 +retrieval&gt;   isa      count-order   first    =num2 !output!  (=num1) )</pre>	<p>De naam van de productie.  Als de goal is,  het tellen vanaf  het nummer =num1  en =num1 is niet het laatste cijfer  en men is bezig te tellen.</p> <p>En een chunk is opgehaald  van het type count-order,  dat vertelt dat =num1  voor =num2 komt.</p> <p>Dan,  verander de goal  om te tellen vanaf =num2.</p> <p>En haal een chunk op  van het type count-order,  die telt vanaf =num2.</p> <p>En geef als output =num1.</p>

Tabel C.3: Een voorbeeld van een fail-productie.

ACT-R	Beschrijving
<pre>(P fail =goal&gt;   isa      is-member   object   =obj1   category =cat   judgment pending =retrieval&gt;   isa      error ==&gt; =goal&gt;   judgment no )</pre>	<p>De naam van de productie.  Als de goal het beoordelen,  van lidmaatschap is  van =obj1  in categorie =cat  en het oordeel is hangende</p> <p>En de chunk retrieval  is gefaald</p> <p>Dan,  verander de goal  en geef een negatief oordeel</p>

De kans op succes van een productie wordt als volgt berekend,

$$P = \frac{Successes}{Successes + Failures} \quad (C.2)$$

'Successes' en 'Failures' zijn het aantal ervaren successen en mislukkingen. Een succes of mislukking vindt plaats wanneer een productie die is bestempeld als een succes of mislukking vuurt. Alle voorgaande producties die bezig waren met het bereiken van dezelfde goal worden dan een succes of mislukking bijgeteld.

Het leren van de kosten, uitgedrukt in tijd, gaat op een vergelijkbare manier,

$$C = \frac{Efforts}{Successes + Failures} \quad (C.3)$$

'Efforts' is hier de totale tijd dat een productie actief was.  $C$  is dus de gemiddelde duur van een regel.

De waarde van de goal kent geen formule, maar is een constante waarde die door ons met de hand wordt toegekend aan een goal.

Tenslotte kan aan de formule nog ruis ( $\sigma$ ) toegevoegd worden,

$$U = PG - C + \sigma \quad (C.4)$$

De ruis wordt gecontroleerd door de utiliteitsruis parameter  $s$ . De ruis kent een verdeling volgens een logistische distributie met een gemiddelde van 0 en een variantie ( $\sigma^2$ ) van

$$\sigma^2 = \frac{\pi^2}{3} s^2 \quad (C.5)$$

### C.3 Declaratief geheugen

Productieregels staan vol met chunks die moeten worden opgehaald uit het geheugen. Chunks bevinden zich in het declaratieve geheugen. Deze kennis correspondeert met dingen waar we ons bewust van zijn dat we ze weten en normaal gesproken kunnen beschrijven aan anderen. Voorbeelden zijn, "George Washington was de eerste president van Amerika" en "Een atoom lijkt op het zonnestelsel".

Procedurele kennis echter, is kennis die tot uiting komt in ons gedrag, maar waar we ons niet bewust van zijn. Niemand kan bijvoorbeeld de regels beschrijven die ervoor zorgen dat we kunnen fietsen of onze moedertaal spreken.

Een chunk ziet er, net als een goal, als volgt uit,

Net als dat er op elk moment meerdere productieregels van toepassing kunnen zijn op de huidige goal, passen er meestal ook meerdere chunks op de plaatsen waar een productie chunks probeert in te vullen.

Tabel C.4: Een voorbeeld van een geheugenchunk.

ACT-R		Beschrijving
fact3+4:		De naam van de chunk.
isa	addition-fact	Het type van de chunk.
addend1	three	En een willekeurig aantal slots,
addend2	four	relaties met andere chunks of slots.
sum	seven	In dit geval drie.

Wanneer meerdere chunks voldoen dan wordt diegene geselecteerd, die de hoogste activatiewaarde heeft. Deze waarde is met de volgende formule gedefinieerd,

$$A_i = B_i + \sum_j W_j S_{ji} + \sum_k P_k M_{ki} \quad (\text{C.6})$$

De sommaties gaan over al de elementen  $j$  in de goal en de elementen  $k$  in de omschrijving waaraan de op te halen chunk moet voldoen.

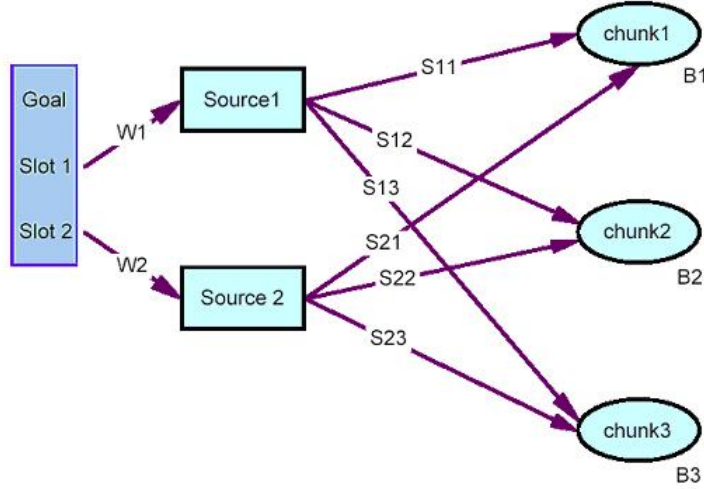
- $B_i$  is de base-level activatie van de chunk  $i$ , een maat voor de sterkte van voorgaande leersessies.
- $S_{ij}$ 's zijn de sterktes van de associaties tussen de elementen  $j$  and  $i$ , tussen respectievelijk de slots in de goal en de chunks.
- $W_j$  reflecteert het gewicht meegegeven aan het  $j$ -de element. Er is een vaste hoeveelheid gewicht wat verdeelt wordt over de elementen van de goal.
- $M_{ki}$  geeft aan hoe goed een element  $k$  van chunk  $i$  overeenkomt met de retrieval beschrijving. Een op te halen chunk kan al gedeeltelijk ingevuld zijn, dit levert extra activatie voor geassocieerde chunks.
- $P_k$ , is een maat voor de nadruk die gegeven wordt aan de overeenkomst met element  $j$ .

In figuur C.3 zien we een diagram waarin activatie vanuit de goalslots terechtkomt bij geassocieerde chunks.

De vergelijking die de base-level activatie beschrijft staat hieronder. Er wordt gesommeerd over het totaal aantal representaties  $n$  van chunk  $i$ , de snelheid waarmee chunks hun activatie verliezen wordt gezet met  $d$ .

$$B_i = \ln \sum_{j=1}^n t_j^{-d} \quad (\text{C.7})$$

Hierin wordt een proces beschreven, waar telkens wanneer een chunk is gepresenteerd zijn base-level activatie toeneemt. Elke activatie neemt bovendien af als



Figuur C.2: goalslots zorgen voor activatie bij chunks.

een exponentiële functie van de tijd. Alle activiteiten veroorzaakt door deze presentaties worden opgeteld en een logaritmische functie levert het eindresultaat. Een presentatie is één van de volgende drie gevallen:

1. de chunk is voor het eerst gemaakt als een ge-pop-te goal. "Popping" is wat gebeurt als de huidige goal wordt bereikt en wordt toegevoegd aan het declaratief geheugen.
2. Wanneer een ge-pop-te goal overeenkomt met een al bestaande chunk wordt het origineel versterkt.
3. Als de chunk ge-retrieve-d is. Dat wil zeggen wordt opgehaald uit het geheugen.

Vergelijking C.7 is echter nogal rekenintensief en er bestaat dan ook een benadering van deze functie waarbij ervan wordt uitgegaan, dat de presentaties gelijk verdeeld zijn over de tijd dat de chunk bestaat<sup>4</sup>. De vergelijking ziet er als volgt uit,

$$B_i = \ln \sum_{j=1}^n t_j^{-d} \approx \log\left(\frac{n}{1-d}\right) - d \log L \quad (\text{C.8})$$

waar  $n$  het aantal presentaties is en  $L$  is de tijd dat de chunk bestaat. De parameter  $d$  is de base level learning parameter en bepaalt de mate van afname, dezelfde functie als in de vorige base-level learning vergelijking.

<sup>4</sup>Het is deze benadering die wij zullen gebruiken in ons model. We bezitten in ons model wel informatie over het aantal representaties van chunks, maar geen informatie over de tijd van deze representaties.

We kunnen ook hier weer ruis toevoegen,

$$A_i = B_i + \sum_j W_j S_{ji} + \sum_k P_k M_{ki} + \sigma \quad (\text{C.9})$$

Ook hier is de ruis weer verdeeld volgens een logistische distributie aangegeven door een parameter  $s$ . Het gemiddelde van de verdeling is 0 en de variantie is,

$$\sigma^2 = \frac{\pi^2}{3} s^2 \quad (\text{C.10})$$

Deze ruis wordt toegevoegd aan elke chunk op elke tijdsstap en ook nog eens aan de retrieval drempel. Als de activatiewaarde van een chunk zich beneden deze drempel bevindt, kan de chunk niet worden opgehaald uit het geheugen.

Tot zover een overzicht van de ACT-R theorie. We hebben hierboven niet alles beschreven wat zij behelst. We hebben uitgelegd hoe productieregels worden geselecteerd maar bijvoorbeeld niets gezegd over hoe ze worden gecreëerd. Voor een volledig overzicht verwijzen we graag door naar Anderson en Lebiere (1998) of de ACT-R research-group website<sup>5</sup>.

---

<sup>5</sup>ACT-R Website, <http://act-r.psy.cmu.edu/>