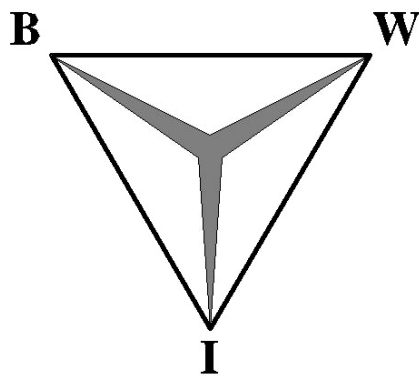


ERVAREN COMPUTERS



*Een introductie van
zelfadaptieve systemen*

BWI-werkstuk, maart 2003
Tjebbe Witteveen

1. Voorwoord

Aan het begin van dit collegejaar was ik op zoek naar een onderwerp voor mijn BWI-werkstuk. Ik wist niet precies waarnaar ik op zoek was. Toen ik echter een onderwerp zag staan naar aanleiding van de zeilboot syllogic, was de keuze snel gemaakt. Om een onderdeel van mijn studie te combineren met een van mijn grote hobby's leek te mooi om waar te zijn.

Het werkstuk gaat niet over zeilen zelf, maar om de techniek die het mogelijk maakt om een zeilboot praktisch zelfstandig te laten zeilen. Deze nieuwe techniek kan mogelijk in de komende jaren voor een doorbraak zorgen en de grenzen die er nu bestaan van wat mogelijk is verleggen. Dit werkstuk gaat over de technieken achter de zeilboot en over mogelijke situaties waarin deze techniek toegepast zou kunnen worden.

Om dit werkstuk te lezen is een begrip van het zeilen niet noodzakelijk, echter om dit werkstuk te schrijven wel. Vandaar dat ik leuk vond om zo mijn hobby en studie te kunnen combineren.

Mijn grote dank gaat uit naar Martijn van Aartrijk van Robosail, het bedrijf dat de techniek achter de zeilboot syllogic ontwikkeld heeft, voor alle nuttige informatie en naar Elena Marchiori, die mij bij het maken van het werkstuk enkele zeer nuttige aanwijzingen gegeven heeft.

Veel plezier bij het lezen,

Tjebbe Witteveen
April 2003

2. Samenvatting

De vraagstelling die als uitgangspunt van dit werkstuk dient is:

Kan de techniek die ontwikkeld is om een zeilboot zelf te laten zeilen, toegepast worden in een bedrijfseconomische omgeving?

Door het bedrijf syllogic is een zeilboot ontwikkeld die praktisch zelfstandig zeilt. Zeilen is geen triviale bezigheid, men kan alleen 'goed' zeilen als men ervaring heeft met zeilen. Er is geen standaard model of algoritme bekend om de het probleem zeilen goed op te lossen. Echter de mens weet wel hoe hij het snelste kan varen, en dat is ook te leren. Deze twee constateringingen scheppen de ruimte waarin zelfadaptieve systemen toegepast kunnen worden. De mens kent de optimale oplossing van het probleem wel, maar er is geen model te maken voor dat probleem.

Aan boord van de syllogic bestaat een hiërarchisch systeem dat de beslissingen neemt. In volgorde de schipper (skipper), de stuurman (navigator), bootsman (watchman) en roerganger (helmsman). Naast deze hiërarchie bestaan er ook verschillende adviserende systemen, onder andere de advisor en de waverider. Al deze systemen krijgen de informatie binnen via sensoren en agents aan boord. Deze systemen beslissen samen over de te varen koers en zorgen dat de boot daar zo snel mogelijk komt.

Agents zijn een belangrijke schakel in het proces. Zij bieden de informatie die door de sensoren ontvangen wordt, pasklaar aan de verscheidene systemen aan. De agents filteren onder andere de ruis eruit en zo zijn zij een onmisbare schakel in dit soort zelfadaptieve systemen. De keuze van de input voor een zelfadaptief systeem zeer belangrijk is daarvoor zijn verschillende technieken bekend, de berekening van de correlatie of een methode op het gebied van datamining.

Er zijn verschillende algoritmen om toe te passen voor het gebruik van zelfadaptieve systemen. De algoritmen aan boord zijn onder te verdelen in drie soorten: k-nearest neighbour algoritmes, een neurale netwerken en ARMA-algoritmen. Deze passen op basis van het verleden zichzelf steeds meer aan en ze optimaliseren zo de snelheid van de boot, de uiteindelijke output waar het om draait.

Ik zie vijf mogelijke situaties waarin deze techniek zijn nut kan bewijzen.

- voorkomen van fouten
- vervangen van mensen
- kennis verspreiden
- meer en sneller gegevens kunnen verwerken
- inzetten van robots in nieuwe situaties

De ontwikkeling van deze systemen staat nog in zijn kinderschoenen. Vele obstakels zullen nog overwonnen moeten worden. Op technisch gebied, de ontwikkeling van algoritmes, maar zeker ook op sociaal maatschappelijk gebied zullen er weerstanden zijn tegen het door robots dingen laten doen die vroeger alleen aan mensen voorbehouden waren. Tot slot wil ik de opmerking plaatsen dat de grens van wat wij computers kunnen leren ligt op het gebied van wat de mens zelf begrijpt.

3. Inhoudsopgave

1. Voorwoord.....	1
2. Samenvatting	2
3. Inhoudsopgave.....	3
4. Introductie.....	4
4.1. Vraagstelling	4
4.2. Wat is (zee)zeilen.....	5
4.3. Opbouw van het werkstuk	6
5. Zelfadaptieve systemen	7
5.1. Wat is een zelfadaptief systeem.....	7
5.2. De syllogie	8
5.3. De zelfadaptieve algoritmen aan boord van de syllogie	10
5.3.1. Knowledge enriched k-nearest neighbour.....	11
5.3.2. Neurale netwerken.....	12
5.3.3. ARMA-model.....	14
6. Situaties waarin een zelfadaptief systeem gebruikt kan worden	15
6.1. Ter vervanging van bestaande functies.....	15
6.2. Situaties die de capaciteiten van mensen overschrijden	15
6.3. De mogelijkheid om kennis te verspreiden.....	16
6.4. De voorkoming van fouten	17
6.5. Nieuwe situaties	17
6.6. ALVINN	18
7. Overwegingen bij het ontwikkelen van een zelfadaptief systeem.....	20
8. Conclusie	22
9. Litteratuurlijst	23

4. Introductie

4.1. Vraagstelling

In het recente verleden is de computer gemeengoed geworden in bedrijven. Men heeft de computer nodig om ingewikkelde berekeningen te maken en problemen op te lossen. Deze technieken hebben allemaal een ding gemeen, ze werken voornamelijk met de rekenkracht en de enorme opslagcapaciteit van de computer. Tegenwoordig kunnen bedrijven niet meer zonder een computer. De computer wordt steeds krachtiger en er zullen dus ook meer mogelijkheden komen. De mogelijkheden van een computer worden echter nog niet ten volle benut. De techniek waarover dit werkstuk staat, staat nog in de kinderschoenen. Er is een enorm potentieel aan mogelijkheden om deze techniek van zelfadaptieve systemen toe te passen.

Een van de gebieden waarop deze techniek ontwikkeld is, is het zeilen. Op dit moment varen er meerdere zeilboten met slechts 1 bemanningslid. Op een van deze boten, de syllogic, ontwikkeld door het bedrijf robosail¹, is deze persoon is met name aanwezig om de daadwerkelijke zeilhandelingen, voor een groot deel op advies van de computer systemen, uit te voeren en eventuele reparaties te verrichten. De rest van de taken wordt verricht door computers. De taken omvatten o.a. het beslissen over de zeilstanden en de zeilvoering², de navigatie, en het sturen van de boot. Het bijzondere aan deze zeilboot is dat hoe meer de boot zeilt hoe meer deze leert en hoe beter de boot gaat zeilen. De software blijft zich constant aanpassen, verbeteren.

De techniek van zelfadaptieve systemen is een recente techniek op het gebied van artificial intelligence. Deze nieuwe techniek roept een boel vragen op bij de mensen die ervan gehoord hebben: wat houdt deze techniek in? Is deze techniek alleen maar een leuk onderzoek of is er ook geld mee te verdienen? Moeten bedrijven in deze techniek investeren? Wat zijn de randvoorwaarden om deze techniek te gebruiken? Uit al deze kleine vragen kwam een centrale vraag naar boven. Dit is de vraag die ik in dit werkstuk wil beantwoorden:

Kan de techniek die ontwikkeld is om een zeilboot zelf te laten zeilen, toegepast worden in een bedrijfseconomische omgeving?

Deze vraag laat zich niet zomaar beantwoorden. Daarvoor zullen we een aantal deelvragen moeten beantwoorden? De deelvragen die ik onderscheiden heb zijn:

- Hoe zeilt de syllogic dan door middel van deze systemen?
- Wat voor technieken zijn er op het gebied van zelfadaptieve systemen?
- Op welke bedrijfseconomische gebieden kan deze techniek gebruikt worden?
- Wat zijn voorwaarden voor en de voor- en nadelen van het gebruik van deze techniek?

Deze vragen wil ik in het werkstuk beantwoorden en de vragen vormen het hart van dit werkstuk. Om de overstap te van een zeilboot naar andere toepassingen te kunnen maken zal ik ter introductie uitleggen wat zeilen is en waarom zeilen niet op een conventionele manier te programmeren is.

¹RoboSail systems B.V. www.robosail.com

²Zeilvoering: welke zeilen gehesen zijn; zeilstanden: de stand van de zeilen, staan de zeilen in het midden van de boot of juist heel ver naar buiten

4.2. Wat is (zee)zeilen

De aanleiding voor dit werkstuk is een zeilboot. Het is handig om enig inzicht te krijgen in wat er aan boord gebeurt en wat zeilen is. Onderstaande is gebaseerd op mijn eigen ervaring op zeilgebied aan boord van “de eendracht³” (een 60 meter lange driemastschoener) en op verscheidene andere boten zoals valken en skûtsjes, voor een uitgebreidere beschrijving van het zeilen verwijs ik naar [2]. Zeilen is voor een gedeelte te leren, maar op zeker moment is het een kwestie van ervaring opdoen. Dezelfde situatie kom je op het water nooit tegen, het blijft af en toe improviseren. Dit geldt voor iedere mens op de hele wereld. Ook bij de wedstrijden op zeilgebied, de Volvo Ocean Race en de America's Cup, komen de bemanningen in onbekende en ongewenste situaties terecht. Zo is er het verhaal van de Illbruck die gigantische windhoos op 100 meter zag passeren, dat hadden ze daarvoor nooit meegemaakt en dat willen ze ook niet nog een keer.

Zeilen is zich met behulp van een zeilboot over het water voortbewegen⁴. Bij wedstrijdzeilen komt er nog een dimensie bij. Je wil zo snel mogelijk, sneller dan de andere boten op een bepaald punt aankomen.

Om zo snel mogelijk ergens aan te komen, moet je gebruik maken van de wind en de zeilen. Als de wind van achteren komt dan zal het niet goed zijn om de zeilen strak in het midden van de boot te zetten, je komt sneller vooruit als de zeilen een open stand hebben. Hoe meer de wind van voren komt hoe strakker de zeilen moeten staan. Recht tegen de wind in zeilen kan geen enkele zeilboot. Dat is lastig, een ander probleem is het sturen van een zeilboot. Doordat op ieder moment de krachten die op de zeilen staan veranderen, de wind en golven veranderen immers ook constant, en brengen de boot van de juiste koers af. Het is lastig om een zeilboot rechtdoor te laten gaan. Het roer staat dus ook niet recht als er rechtdoor gevaren wordt. Het roer dient om de krachten die op het schip staan te corrigeren om uiteindelijk rechtdoor te varen.

Deze ‘problemen’ zijn voor de mensen aan boord redelijk makkelijk op te lossen. Er wordt op een punt op de wal of op een bepaalde kompas koers gevaren. Op een kompas koers varen is nog te kwantificeren. Stel de gewenste koers is 45° en de huidige koers is 55° dan zal er naar bakboord (links) gedraaid moeten worden. En wel exact 10°. Daarna zal de boot weer van koers afraken en met dat weer gecompenseerd worden. Het gebruik van het roer moet zoveel mogelijk voorkomen worden, want elke roerbeweging werkt als een rem dus een stabiele roerganger levert snelheidswinst op.

Lastiger is het ‘oploeven⁵ in een vlaag’. Dit is mogelijk als de boot ‘aan de wind⁶’ vaart, omdat de wind even harder gaat waaien. Het heeft te maken met het feit dat de bootsnelheid zich nog niet aangepast heeft aan de nieuwe windsnelheid, waardoor de wind tijdelijk ruimer inkomt. Hoeveel graden er precies opgeloeft moet en kan worden zal in elke situatie verschillen, er zijn geen getallen voor te geven. Geen zeiler kan exact vertellen hoeveel graden de boot moet oploeven in een vlaag. Dat is afhankelijk van de duur, sterkte van de vlaag, van het soort boot etc. Op een groot zeezeilschip als de Eendracht, reageert het roer te langzaam om er werkelijk op te kunnen reageren, op een laser zal de boot omslaan als er niet op de vlaag geanticipeerd wordt. Zo groot zijn de verschillen. Het is onmogelijk om dit aan een systeem op te geven wat er moet gebeuren, een regel als “als er een vlaag is loef dan 5 graden op” is onmogelijk. Het is een kwestie van ervaring: voelen wanneer de boot weer recht in het water komt te liggen, voelen wanneer de druk op het roer vermindert. Hier komen de problemen echt om de hoek kijken, hoe kan je iets wat niet in getallen uit te drukken is netjes

³ www.eendracht.nl

⁴ Van Dale online (www.vandale.nl)

⁵ Oploeven is naar de wind toedraaien

⁶ ongeveer onder een hoek van 45° ten opzichte van de werkelijke wind

programmeren? Hoe kan je anticiperen op de steeds veranderende weeromstandigheden? Het is onmogelijk om elke situatie apart te programmeren.

Dit zijn slechts enkele aspecten van het zeilen, om duidelijk te maken dat zeilen, en zeker wedstrijdzeilen, geen gemakkelijke bezigheid is. Zeilen is echt een ervaringsbezigheid, hoe meer je het doet hoe beter het zal gaan. Het is een complex spel van wind, weer, golven, zeilstanden en reageren op wat er gebeurt. Zeker op de oceaan zal je de weerverwachtingen goed moeten bestuderen om de juiste beslissing te nemen. Ook kan je gebruik maken van de golven die er zijn. Dit geheel is niet in een systeem van IF... THEN... regels te beschrijven. Afhankelijk van de omstandigheden zal je andere beslissingen nemen.

Ik heb een gedeelte van de problematiek van het zeilen proberen aan te geven en aan te geven dat zeilen op zee lastig is. Een volledige uitleg is onmogelijk en valt ook buiten het doel van dit werkstuk. De enige manier om dit goed aan te voelen is om zelf een keer in een bootje te gaan zitten en het te ervaren.

4.3. Opbouw van het werkstuk

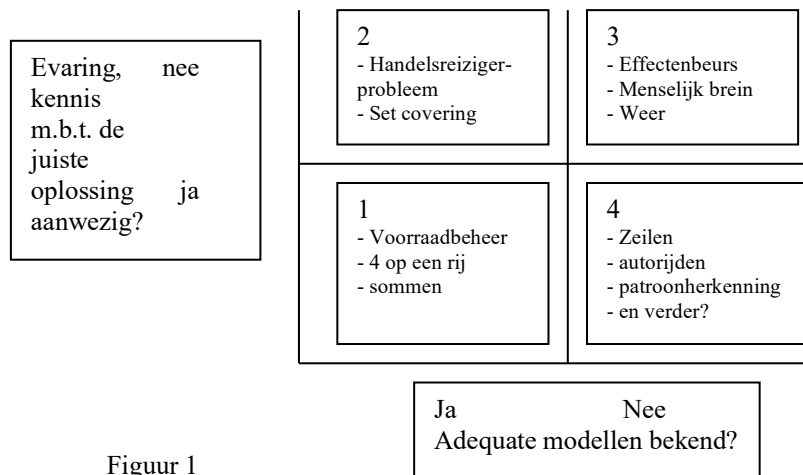
Het mogelijk maken dat een zeilboot zelfsturend wordt is, zoals beschreven, een lastig probleem. Toch is dat gelukt. Dat is al een prestatie op zich. De weg van een zelfsturende zeilboot naar bedrijfseconomische toepassingen van deze zelfadaptieve techniek is nog lang. Dit werkstuk is daar slechts een beginnende aanzet toe.

Om uiteindelijk op de vraagstelling een antwoord te geven zal ik in mijn werkstuk de volgende opbouw hanteren. Om te beginnen zal ik aangeven wat een zelfadaptief systeem is en waarom een zelfadaptief systeem gebruikt kan worden om een boot te leren zeilen. Daarna zal ik het zeilen aan boord van de syllogic bespreken, vervolgens de verschillende algoritmes die gebruikt worden om te syllogic te laten zeilen beschrijven. Tot slot wil ik aangeven hoe deze techniek in de toekomst op bedrijfseconomisch gebied gebruikt kan worden en wat de voor- en nadelen van deze techniek zijn.

5. Zelfadaptieve systemen

5.1. Wat is een zelfadaptief systeem

Blijkbaar kunnen zelfadaptieve systemen gebruikt worden om een boot te leren zeilen, maar waar moeten we verder aan denken? Welke andere toepassingen zouden er kunnen zijn? In dit hoofdstuk wil ik uitleggen wat een zelfadaptief systeem is en waar we aan moeten denken bij deze systemen. De basis voor deze uitleg vormt figuur 1⁷.



Figuur 1

Op de twee assen staan twee vragen. Vraag 1: “zijn er adequate modellen bekend?” Vraag 2: “Is er kennis en ervaring m.b.t. de juiste oplossing aanwezig?”. Door deze twee vragen met ja en nee te beantwoorden komen we uit in een van de vier kwadranten. Deze vragen verdienen enige toelichting. Bij de eerste vraag gaat het erom of we een model kunnen maken van het probleem en dan een oplossing of een goede benadering van deze oplossing kunnen geven. Bij de tweede vraag gaat het om het feit of wij als mensen weten wat de juiste oplossing is. Van een simpele som weten we dat natuurlijk. Maar ook bij bijvoorbeeld zeilen weten we wat we moeten doen, komt de wind van achteren dan moet het zeil breed uitstaan om de maximale snelheid te halen. Bij het besturen van een auto weten we dat we moeten stoppen bij een rood stoplicht.

Zijn de modellen bekend en weten we wat de juiste oplossing zou moeten zijn, dan komen we uit in het eerste kwadrant. Te denken valt hierbij aan bijvoorbeeld een spel als “4 op een rij” waarvan de optimale strategie bepaald is, of een simpele optelsom. De kracht van de computers is dat ze de berekening veel sneller kunnen doen en daardoor zijn ook steeds moeilijker berekeningen op te lossen. Voor een mens is het mogelijk om voor de computer een programma te schrijven wat er uit gerekend moet worden. Daarna is het een kwestie van rekenkracht. Er is geen creativiteit meer nodig.

Zijn de modellen wel bekend maar de juiste oplossing niet, dan komen we in het tweede kwadrant. Hierbij valt te denken aan handelsreizigersproblemen, of een set-covering probleem⁸. We weten dat we bij benadering goed zitten, we weten ook ongeveer hoe ver we van de optimale oplossing afzitten. Het is meestal mogelijk om een model te schrijven dat tot een optimale oplossing komt, maar het duurt te lang om deze oplossing ook werkelijk te kunnen bepalen. Immers tijd om de exacte oplossing te berekenen neemt niet polinomiaal toe.

⁷ Dit schema werd mij aangereikt door P. Adriaans

⁸ Zie voor deze problemen [10]

In het derde kwadrant zijn de modellen niet bekend en we weten ook geen adequaat model om het probleem te beschrijven. Te denken valt aan het weer dat ook de weermensen nog steeds voor verrassingen stelt of de werking van ons menselijk brein. Misschien dat we ooit nog eens tot een oplossing van deze problemen zullen komen, maar zolang als we zelf niet weten hoe we het eigenlijk moeten doen kunnen we het ook niet aan computers leren.

In het vierde kwadrant zijn we gekomen tot de situatie dat we zelf als mens goed weten wat we moeten doen maar dat we dat niet kunnen modelleren. Natuurlijk kunnen we wel ongeveer aangeven wat er moet gebeuren maar juist die fine-tuning is te bepalen. Zoals aangegeven is zeilen daar een voorbeeld van, maar bijvoorbeeld ook autorijden of vliegen. Deze situaties kunnen niet geleerd worden aan computers. Hier is een nieuwe techniek voor nodig, namelijk die van zelfadaptieve systemen.

Dit brengt mij op een definitie van een zelfadaptief systeem. Deze luidt als volgt:

Een zelfadaptief systeem is computergestuurd systeem ontworpen om een bepaalde taak uit te voeren; dit systeem leert zichzelf door middel van een terugkoppeling deze taak steeds beter uit te voeren.

De kracht van deze systemen zit vooral in het steeds maar door blijven leren, ze worden steeds beter. Er gaat wel een redelijk lange tijd aan vooraf voordat de systemen de taak goed uitvoeren en er zal ook redelijk veel kennis meegegeven moeten worden aan het systeem om het uiteindelijk de taak goed te laten uitvoeren. Het voordeel van kennis aan het systeem meegeven is dat het algoritme veel sneller in staat is zijn taak goed uit te voeren.

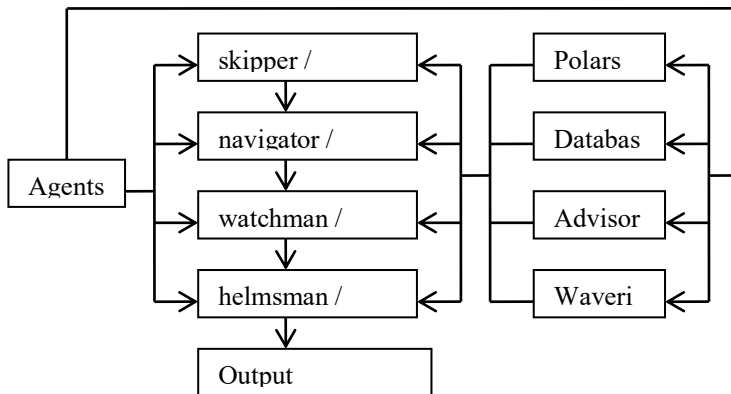
Een kort illustratief zeilvoorbeeld. De boot vaart een bepaalde koers recht op het eindpunt af en het doel is daar zo snel mogelijk aan te komen. Op een gegeven moment draait de wind en komt de boot te veel tegen de wind in te varen zodat de boot een stuk langzamer gaat varen. Nu kan de boot door blijven varen of twee kanten op draaien van de wind af of naar de wind toe. Er zal een beslissing genomen moeten worden. Het is noodzakelijk om in eerste instantie een stukje van de wind af te draaien om weer vaart te krijgen. Tegen de wind in varen lukt nou eenmaal niet. Daarna moet afhankelijk van de windrichting en het doel op een gegeven moment door de wind heen gedraaid worden. Dit kan je meegeven aan een systeem, als (varend op een koers van 45° ten opzichte van de wind) en (de wind draait zodat de hoek $< 45^\circ$) dan (draai van de wind af). Heel belangrijk is het dus dat het systeem 'weet' wat van de wind af draaien is. Als het systeem dat niet weet dan is deze situatie lastig te verhelpen. Hoeveel graden er van de wind af gedraaid moet worden dat zijn juist dingen die het systeem zelf kan leren maar de kennis om van de wind af of naar de wind toe te draaien is essentieel om zeilen te leren. Deze kennis moet dus wel van tevoren aan het systeem meegegeven worden. Een zelfadaptief systeem is dus een systeem waarin ook een heleboel kennis gestopt moet worden, de zelfadaptieve systemen kunnen pas vanuit die kennis verder leren. Je kan wel 4 keer rond de wereld varen maar het is de vraag of het systeem dan kan zeilen. Hoe meer kennis het systeem al heeft hoe sneller het zijn taak leert.

5.2. De syllogic

Om te laten zien hoe de syllogic zeilt en wat de overeenkomsten zijn tussen de systemen aan boord van de Eendracht en aan boord van de syllogic wil ik graag een voorbeeld beschouwen. De reis die ik als voorbeeld wil nemen was de reis van Brest naar Vlissingen die ik april 2002 gemaakt heb. We zijn via de eilanden Ouessant, Brehat, Jersey, Sark en Wight naar de haven van Oostende gevaren met als eindbestemming Vlissingen. De schipper heeft de bestemming bepaald. De stuurman heeft ervoor gezorgd dat er veilig naar de bestemming

gevaren werd en dat het doel bereikt werd. De bootsman zorgde voor de zeilstanden en de roerganger stuurde de boot op de aangegeven koers. Ondertussen werd er uitkijk gehouden door de verschillende passagiers om te zorgen dat we nergens tegenaan voeren. Zo kwamen we veilig in de haven van Vlissingen aan.

De zeilboot de syllogie zeilt op basis van verschillende samenwerkende zelfadaptieve systemen. In onderstaande figuur (figuur 2) is aangegeven hoe de samenhang / hiërarchie is tussen de verschillende systemen.



Figuur 2; de systemen aan boord van de syllogie



Figuur 3; de syllogie

Dit schema zal ik nu nader toelichten, eerst de systemen die daadwerkelijk met het zeilen te maken hebben, daarna de adviseurs van die systemen. Het ontwerp dat door de makers van robosail gekozen is komt overeen met de functies die aan boord van een door mensen bemand schip. Ze hebben gekeken hoe het aan boord was en geprobeerd deze situatie door middel van computers na te bootsen.

Aan boord is de schipper altijd de baas de eindverantwoordelijke, de schipper bepaalt waar de boot heen gaat en via welke weg die daar zal komen. Daarnaast zal de schipper ingrijpen op het moment dat het niet goed gaat of op het moment dat het schip in zware, lastige omstandigheden terechtkomt. De schipper houdt zich niet bezig met de korte of middellange termijn, hij is vooral bezig met de lange termijn. Met name de routeplanning en dan rekening houdend met weer en getij. Aan boord zie je de schipper niet of nauwelijks voor de mensen die meevaren is de schipper, zeker op grote boten, nauwelijks zichtbaar. Echter als het moet is hij er 16 uur of langer aan een stuk door.

De stuurman heeft de leiding als de schipper afwezig is, deze bepaald op de middellange en korte termijn wat er moet gebeuren. Moet er uitgeweken worden voor andere schepen, moet de koers bijgesteld worden, hoe is de stroming. Kortom de stuurman zorgt dat de positie bijgehouden wordt en voor alle navigatie. Daarnaast is het een ook een controle functie, continue is hij bezig om te kijken of alles goed gaat. De stuurman is een soort van plaatsvervangende schipper.

Bij de bootsman begint het echte zeilende werk, hij gaat namelijk over het zeilende gedeelte van de boot. Elke zeg 10 seconden bekijkt dit systeem of er iets met de zeilen moet gebeuren of de zeilen goed staan of er van zeil gewisseld moet worden of de zeilen kleiner moeten worden. Dat doet het systeem onder andere door de huidige situaties te vergelijken met situaties die het in het verleden ervaren heeft.

De roerganger zorgt ervoor dat het schip de juiste koers vaart, dat is in een auto misschien makkelijk maar met zeilen is dat nog erg lastig. De wind varieert constant in kracht en

richting en rechtdoor varen betekent zeker niet het roer recht houden. Het roer probeert de krachten die op het schip staan te corrigeren om zo de juiste koers te varen en soms juist gebruik te maken van deze veranderende omstandigheden. Elke 3 seconden kan de roerstand weer aangepast worden.

Verder zijn er aan boord een soort van adviseurs van deze systemen, zij ondersteunen de vier basisfuncties. De polars bestaan uit gegevens geleverd door de ontwerper van de boot. Bij het ontwerp is uitgerekend wat bij iedere windsnelheid en koers ten opzichte van de wind de optimale snelheid zou moeten zijn. Dit is echter een ideaal plaatje, in de praktijk is het zo dat als je op 75-80 procent van de optimale snelheid vaart je behoorlijk goed is. De Waverider heeft met name betrekking op de roerganger, deze kan de koers die de roerganger moet varen aanpassen aan de golven die er zijn. Het is zo dat de golven tegen je kunnen werken op zee, maar je kan ze ook gebruiken. Even iets langer met een golf meevaren kan voordeel opleveren. In de database staan de gegevens die in de loop der tijd verzameld zijn opgeslagen, deze geven onder andere de advisor inzicht in welk advies er gegeven moet worden. Mocht er ooit een met dezelfde windsnelheid en richting sneller gevaren zijn dan er op dat moment gedaan wordt, dan moet er dus duidelijk iets veranderd worden. Daarnaast bestaat er ook nog een soort schema van de optimale snelheid van de boot bij bepaalde weersomstandigheden. Dat zijn de polairen van de boot. Ook hiermee wordt de snelheid van de boot vergeleken om te bepalen of de boot nog harder kan.

Tot slot van de uitleg van deze figuur wil ik nog de agents bespreken. Dit zijn een soort relevante informatie verzamelaars en doorgeef kanalen. Aan boord komen, door middel van sensoren, iedere seconde heel veel gegevens binnen. Er moet dan gedacht worden aan gegevens over de golfhoogte, andere schepen in de buurt, windsnelheid en windrichting enzovoort. Als al deze gegevens aan de systemen doorgegeven zouden worden dan zouden deze systemen meer tijd kwijt zijn aan het verwerken van die gegevens dan aan het werkelijk uitvoeren van hun taak. Daarom bestaan agents. Zij ontvangen de gegevens van de sensoren en zetten deze om in informatie die ze aanbieden aan de onderliggende systemen.

Door middel van deze deelsystemen is het gelukt om een zeilboot te laten zeilen. Duidelijk is de samenwerking tussen de verschillende systemen en de opsplitsing van de taken. Daarnaast is de opsplitsing in kleinere deelgebieden belangrijk. Elk van deze kleine delen is goed te begrijpen en goed te programmeren. Opsplitsen is dus een van de kernbegrippen bij het maken van een zelfadaptief systeem. Een van de hulpmiddelen daarbij kan de vakkennis van mensen zijn, soms is het gebied al als vanzelf in delen opgesplitst zonder dat de mensen het zelf door hebben. Autorijden kan bijvoorbeeld opgedeeld worden in het sturen, het schakelen, het controleren van de snelheid, het kijken in 4 (of meer) richtingen, het plannen van de route en nog meer.

Nu gaan we terug naar het voorbeeld aan het begin van dit hoofdstuk. Op de syllogic zou het ongeveer hetzelfde gegaan zijn. De schipper bepaalt de bestemming, de lange termijn planning. De stuurman zorgt voor de korte termijn planning zodat er veilig gevaren wordt. De bootsman zorgt voor de zeilstanden en de roerganger stuurt de boot op de aangegeven koers. Ondertussen werd er uitkijk gehouden door de verschillende sensoren en werden de gegevens door agents verwerkt, om te zorgen dat we nergens tegenaan voeren. Zo kunnen we zien dat aan boord van de syllogic eigenlijk dezelfde functies vervuld worden als aan boord van een regulier schip. De taakomschrijvingen van de verschillende systemen / functies kan ook bijna een op een overgenomen worden.

5.3. De zelfadaptieve algoritmen aan boord van de syllogic

Er zijn aan boord van de syllogic 3 verschillende soorten zelfadaptieve systemen te vinden: een knowledge enriched K-nearest neighbour (verder kNN) algoritme, een neurale

netwerk en een ARMA-model. Daarnaast zijn er nog andere algoritmes te bedenken die gebruikt zouden kunnen worden als zelf adaptief systeem, maar ik wil de algoritmen die gebruikt worden nader toelichten. Deze systemen zijn allemaal gemaakt om de fine-tuning te regelen juist de regels die niet te kwantificeren zijn vorm te geven. Naast deze systemen zijn er bepaalde kennissystemen waarin de regels opgenomen zijn. Dus (als van de wind afgedraaid wordt) dan moet het zeil gevierd worden, maar hoeveel dat wordt door een zelfadaptief systeem bepaald. Ik bespreek de algoritmen hier slechts in globale termen omdat iedere situatie weer om zijn eigen implementatie van deze algoritmen zal vragen. Het basisidee achter het algoritme is in deze belangrijker dan de formules en alle details en op en aanmerkingen, daarvoor verwijs ik naar de literatuur met name Mitchell.

5.3.1. Knowledge enriched k-nearest neighbour⁹

De basis van een kNN algoritme is een kennis over het onderwerp zelf. Eigenlijk alle basisregels staan in een of meerdere kennissystemen. Een aantal voorbeelden: IF (vlaag en koers aan de wind) THEN (loef op(loefwaarde)); IF (draai van de wind af) THEN (zet zeilen ruimer(trimwaarde)); bij auto's: IF(Remlichten zichtbaar) THEN (verminder vaart(remwaarde)). Zo zouden er op elk gebied heel veel regels op te stellen zijn om een computersysteem te laten weten wat je er zou moeten gebeuren. We weten de richting waarin we moeten reageren, alleen we kunnen niet kwantificeren hoever we in deze richting moeten gaan. Dat baseren we op gevoel op intuïtie. We weten niet hoeveel we op moeten loeven, niet weten hoeveel ruimer de zeilen moeten staan, hoe hard we moeten remmen. Dat hangt van een groot aantal factoren af.

Bij een kNN komen al deze factoren op verschillende assen te staan, zodat er een n-dimensionale ruimte ontstaat. Daarna worden verschillende voorbeeldsituaties in deze ruimte gemarkeerd zodat een grote puntenwolk ontstaat.

Stel er is een nieuwe situatie $x := (a_1(x), a_2(x), \dots, a_n(x))$ waarvan we willen weten wat er dan moet gebeuren. Dan definiëren we de afstand van x tot de rest van de punten als $d(x_i, x_j) \equiv \sqrt{(\sum_{r=1}^n (a_r(x_i) - a_r(x_j))^2)}$, de standaard euclidische afstand. Dan is de dichtstbijzijnde buur de het punt waarvoor deze afstand minimaal is. De k in kNN is het aantal burens wat we meenemen om de klasse van x te bepalen. Als $k=1$ dan nemen we alleen de dichtstbijzijnde buur in beschouwing, als $k=151$ de dichtstbijzijnde 151 burens. Het aantal burens zal natuurlijk afhangen van de hoeveelheid data en hoe goed de data zijn. Het kan ook nog zijn dat we van de k dichtstbijzijnde burens een gemiddelde waarde nemen. Op basis van dit principe zijn veel verschillende variaties mogelijk. We kunnen door toevoeging van gewichten het ene attribuut belangrijker maken dan de andere.

Op deze manier kunnen we juist die kleine gevoelsmatige dingen aan een computer aanleren. Op de syllogic is men gaan varen met de boot en wordt elke actie opgeslagen in een grote database. Zo ontstaan vanzelf tientallen voorbeelden van oploeven in een vlaag. Maar voor de andere voorbeelden kan het net zo gelden. En dan is het mogelijk dat de computer uiteindelijk zelf doorheeft wat er gedaan moet worden.

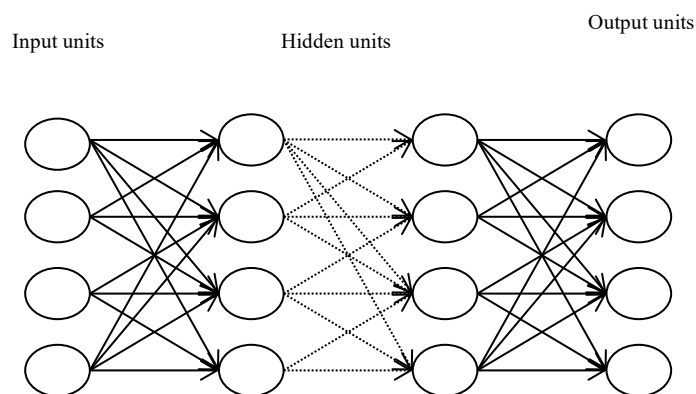
Dit algoritme is zelflerend omdat iedere nieuwe situatie een verandering van het algoritme aanbrengt. Elke nieuwe situatie kan namelijk toegevoegd worden aan de data. Echter het is wel belangrijk deze goed te controleren zodat alleen de voorbeelden die een positieve bijdrage hadden toegevoegd worden. Door de slechte acties weg te laten blijft uiteindelijk een goed functionerend algoritme over dat zijn taak naar behoren uitvoert. Aan boord van de syllogic zijn o.a. de bootsman en de advisor op een knowledge enriched k-nearest-neighbour netwerk gebaseerd. Er wordt dan door deze systemen met name naar de richting van (schijnbare) wind en naar de zeilstanden en zeilvoering gekeken. Is er met deze windsnelheid en -richting ooit

⁹ Dit hoofdstuk is geschreven op basis van hoofdstuk 8.2 van Mitchell

sneller gevaren dan moeten we kijken hoe deze zeilstanden aangepast moeten worden om weer die snelheid te behalen. Alles is er natuurlijk op gericht om de boot zo snel mogelijk door het water te laten gaan.

5.3.2. Neurale netwerken¹⁰

Een andere techniek die gebruikt kan worden zijn neurale netwerken. Deze techniek bestaat uit een netwerk met verschillende lagen, een input laag, een of meerdere verborgen lagen en een output laag. Belangrijk is dat de output als een discrete vector gemodelleerd moet kunnen worden. Er zijn maar een eindig aantal acties die gedaan kunnen worden. Op basis van een bepaalde input wordt de output, de actie die gedaan moet worden, bepaald. De output is de actie die ondernomen moet worden. De roerganger aan boord is met deze techniek uitgerust. Zoals eerder gezegd een zeilboot vaart eigenlijk nooit rechtdoor er moet altijd voor de kracht die op de zeilen staat gecompenseerd worden. Welke kant de boot op wil is het best te meten door de druk die op het roer staat te registreren.



Figuur 4; een neuraal netwerk

Het principe klinkt heel simpel maar de praktijk is lastiger. Bij elke knoop wordt aan de input van die knoop een bepaald gewicht meegegeven, dat geldt voor de input, hidden en output knopen. Op basis van trainingsdata, waarbij bij elke input de gewenste output bekend is, worden deze gewichten aangepast. Het netwerk moet op basis van deze data getraind worden om de taak uit te voeren.

In elke knoop wordt door de input te vermenigvuldigen met een vector van gewichten een bepaalde waarde berekent. Door deze waarde de input te laten zijn van een bepaalde functie (bijvoorbeeld de sigmoid functie¹¹), wordt de output van deze knoop bepaald. De waarde van de output komt in meer of mindere mate overeen met de gewenste output. Om het verschil tussen in- en output kleiner te maken worden de gewichten waarmee de input vermenigvuldigd wordt aangepast. Dit kan op basis van verschillende algoritmen. Een van de bekendere algoritmen is het backpropagation algoritme¹². Dit algoritme werkt met het verschil van de werkelijke output van het netwerk en de gewenste output van het netwerk. Deze fout wordt als volgt gedefinieerd:

$$E := \frac{1}{2} * \sum_{(d \in D)} \sum_{(k \in \text{outputs})} (t_{kd} - o_{kd})^2$$

waarbij D de set van trainingsvoorbeelden is; outputs de verzameling van outputknopen; en t en o de gewenste en werkelijke output van deze knoop k, bij trainingsvoorbeeld d. Daarna wordt op basis van deze fout de waarde van de gewichten aangepast. Deze berekening van de

¹⁰ geschreven op basis van hoofdstuk 4 van Mitchell

¹¹ de sigmoid functie: $\sigma(y) = \frac{1}{1+e^{-y}}$

¹² Hoofdstuk 5 van Mitchell

fout gebeurt voor de gewichten van iedere knoop in het netwerk zo wordt het netwerk dus constant aangepast en ‘beter’.

De roerganger die aangestuurd wordt door een neurale netwerk, krijgt de volgende vector van input mee:

- schijnbare windsnelheid/-richting
- bootsnelheid (SOW)
- kompas (COW)
- WaveRider
- GPS koers (COG) en snelheid (SOG)
- huidige roerhoek
- stroomverbruik roer
- temperatuur rudder drive
- boothelling

Hieronder zal ik uitleggen wat de verschillende inputwaarden inhouden en waarom ze aan het neurale netwerk meegegeven moeten worden.

De schijnbare windsnelheid is de snelheid die op de boot gemeten wordt, door de snelheid van de boot wijkt deze af van de werkelijk windsnelheid. Hetzelfde geldt voor de schijnbare windrichting. Dat deze waarden meegegeven moet worden lijkt me erg logisch, de wind is de basis om te kunnen zeilen.

De bootsnelheid (speed over water) is van belang omdat bij verschillende bootsnelheden de boot sneller of langzamer reageert. In het meest extreme geval, als de boot helemaal geen snelheid heeft, zal de boot helemaal niet op een roeruitslag reageren, het heeft dan ook geen zin om iets met het roer te doen.

De kompasskoers (course over water) die gevaren moet worden, moet natuurlijk ook meegegeven worden.

De waverider (zie §5.3.3) projecteert de boot op de niet vlakke zee oppervlakte en zorgt dat juist van de golven gebruik gemaakt wordt.

De COG (course over ground) SOG (speed over ground) worden natuurlijk ook meegegeven dit zijn de huidige koers en snelheid over de grond van de boot.

Dat de huidige roerhoek ook meegegeven wordt is ook erg logisch, er moet bekend zijn welke kant de boot nu opgestuurd wordt. Moet er verder gedraaid worden of moet de draai juist minder hard voortgezet worden.

Het stroomverbruik staat voor de kracht die er op het roer staat. Wil de boot uit zichzelf rechtdoor varen of juist naar de wind toe of van de wind af (is de boot loef- of lijgierig)

Tot slot de boothelling: iedere boot vaart onder een bepaalde helling het snelste. Die hoek hangt ook weer af van de koers ten opzichte van de (schijnbare) wind.

Met deze inputdata wordt dan door het neurale netwerk de output, de roeruitslag, berekend. Dit is een soort van black box voor de eindgebruiker, maar het lijkt logisch dat op basis van deze input een goede output te voorspellen is.

Van sommige van deze zaken wordt een al dan niet korte geschiedenis bijgehouden, alsmede wat statistieken over de variatie van de data e.d. Deze worden gebruikt om het gedrag aan te passen (onder andere de hoeveelheid roerbewegingen minimaliseren), en op een hoger niveau soms gebruikt om te kijken of sensordata nog wel correct is. Verder is er intern binnen de verschillende deelsystemen veel feedback, om te zien of wat er gebeurt wel werkt. Ook gaat soms wat uitvoer naar het computerscherm toe, om een gebruiker te informeren of te waarschuwen. Zo werkt het systeem van de roerganger aan boord van de syllogic.

5.3.3. ARMA-model

ARMA-model staat voor auto regressive, moving average model en het wordt gebruikt om door middel van een terugkoppeling een bepaald evenwicht instant te houden. Dat gebeurt op basis van tijdreeksen. De kenmerken van een tijdreeks zijn dat de volgorde van de waarnemingen belangrijk is. Het domein is oneindig, dat wil zeggen dat er geen eindpunt te bepalen is. En tenslotte moet de reeks stationair zijn, het maakt niet uit wanneer we beginnen met naar het model te kijken we zien op elk moment hetzelfde verschijnsel. Deze modellen worden bijvoorbeeld veel bij helikopters gebruikt om de piloot te helpen.

Bij een AR-model wordt het verleden geanalyseerd; er moet gedacht worden aan het feit dat een object zich rond een bepaald gemiddelde beweegt en dat de afwijking van dat gemiddelde bepaald wordt door een bepaalde random fout. p is het aantal waarnemingen uit het verleden dat meegenomen wordt in de berekening.

$x_t - \mu = \varphi_1(x_{t-1} - \mu) + \dots + \varphi_p(x_{t-p} - \mu) + E_t$ (waarbij μ staat voor de gemiddelde van x_t in het proces en φ voor een ruis en E_t voor de fout die gemaakt wordt op tijdstip t)

Een MA model is een model dat probeert te voorspellen wat er in de toekomst gaat gebeuren. Eigenlijk is dit een zelfde soort model. We nemen aan dat het object zich rond een bepaald gemiddelde bevindt en dat de afwijkingen een soort van random fout zijn.

Door deze te koppelen proberen we op basis van het verleden de toekomst te bepalen. Hoever we daarvoor terug in het verleden gaan en hoever we de toekomst willen voorspellen wordt bepaald door de parameters p en q .

Het ARMA (p,q) model is als volgt gedefinieerd.

$$y(k) = \sum_{(i=1\dots p)} a_i y(k-i) + \sum_{(j=0\dots q)} b_j u(k-j)$$

Er ontstaat alleen een AR model als $q = 0$ en er ontstaat een MA-model als $p=0$. Dit model is bruikbaar als er een bepaalde evenwichtsituatie nagestreefd. Er is bekend dat die evenwichtsituatie er is en men probeert nu de afwijkingen ervan te corrigeren door te kijken naar wat er in het recente verleden gebeurt is. Op een boot wil je dat die zo stabiel mogelijk ligt hoe minder roeruitslag en roerbeweging hoe harder de boot vaart. Echter dat evenwicht wordt continue verstoort door de krachten van water en wind die op de boot staan. Door dit model probeert men de boot zo stabiel mogelijk te houden. Aan boord wordt een ARMA(30,0) model gebruikt in het systeem van de waverider. Deze kan invloed uitoefenen op de roerganger. Het is namelijk niet goed om elke koersverandering volledig te compenseren. Soms raakt de boot van koers door een hoge golf maar dan komt de boot aan de andere kant van de golf weer op koers. Dan wordt er juist eerst teveel naar de ene kant gestuurd en dan teveel naar de andere kant. Zo krijg je juist meer roeruitslag en gaat de boot langzamer. Soms moet je een boot gewoon even laten gaan. Dit model is met name van toepassing als er bepaalde evenwichtsituaties zijn of gehandhaafd moeten worden. Overigens zijn er ook andere adaptieve algoritmen in omloop zoals blijkt uit verschillende artikelen die op het internet te vinden zijn. (zie daarvoor o.a. [3])

Dit zijn slechts drie technieken die men toe kan passen om een systeem zelflerend te maken. Door kennis te verzamelen over een bepaald probleem en die situatie te analyseren kunnen er verscheidene variaties en andere algoritmen ontwikkeld worden die in andere omstandigheden toepasbaar zijn.

6. Situaties waarin een zelfadaptief systeem gebruikt kan worden

Tot nu toen heb ik het alleen maar gehad over de zeilboot syllogie heb ik uitgelegd wat zelfadaptieve systemen zijn. Nu wil ik laten zien dat deze techniek ook in het bedrijfsleven van toepassing kan zijn. Op dit moment wordt er door verschillende bedrijven, onder andere NASA, al gebruik gemaakt van deze techniek maar het is nog geen gemeengoed. Volgens mij zijn er een aantal mogelijke situaties om deze techniek toe te passen maar de precieze invulling zal per bedrijf en per situatie verschillen. Hieronder wil ik deze denkrichtingen aangeven waarbij het mogelijk zou kunnen zijn om de techniek van zelfadaptieve systemen toe te passen. Bij elk van deze situaties zal ik een voorbeeld geven om de situatie duidelijk te maken. Het is goed om tot voorzichtigheid en realisme te manen. Sommige van de situaties die beschreven worden, zijn niet op erg korte termijn realiseerbaar, over 30 jaar zal het misschien tot de mogelijkheden behoren.

6.1. Ter vervanging van bestaande functies

Op een gegeven moment zijn de computersystemen zo goed, dat ze werkelijk de taken over kunnen nemen. Te denken valt aan situaties die sommige personen moeten ontlasten en dus ruimte geven om andere nuttiger zaken te doen. Ook is het natuurlijk mogelijk om zo mensen compleet te vervangen. De vraag is of dat wenselijk is, de kennis van een computer is natuurlijk eenzijdiger en minder creatief dan die van de mens. Het is ook de vraag of het veilig is om echt alleen maar computers te laten werken, juist door hun beperkte kennis kunnen ze goed overzien wat het beste zal zijn. Het zal dus gaan om duidelijke taken die onder standaard omstandigheden prima door een zelfadaptief systeem te volbrengen zijn.

Een voorbeeld is autorijden. Onder standaard omstandigheden en op de snelweg zou deze taak prima door een zelfadaptief systeem te volbrengen zijn. Dat blijkt ook uit het ALVINN project (zie § 6.6). Iedereen die kan autorijden weet dat het een complex is van kijken en met die informatie de juiste acties ondernemen. Er zijn sensoren of agents te maken die de afstand tot de voorganger kunnen bepalen, die achteruit kunnen kijken, kortom die alles kunnen waarnemen wat de mens met zijn zintuigen kan waarnemen.

Een ander voorbeeld kan zijn het bezorgen van de post. Neem een robot met voor elke straat / wijk een opbergsysteem. Daarnaast zal deze robot moeten leren hoe zich op straat te bewegen, uit te kijken voor rijdende en stilstaande objecten. Ook zal het systeem de brievenbus moeten kunnen vinden en er zal een manier gevonden moeten worden dat de brieven daadwerkelijk gepost kunnen worden.

Het tweede voorbeeld is lastiger dan het eerste omdat de diversiteit in het tweede geval groter is. Niemand heeft precies dezelfde tuin, is de brievenbus op dezelfde plek geplaatst, terwijl over het algemeen de snelwegen redelijk op elkaar lijken. Het zal veel lastiger zijn om een auto ook in woonwijken te laten rijden. Bij het eerste geval is de taak ook het probleem dat opgelost moet worden, namelijk autorijden. In het tweede geval is het bezorgen van de post slechts een deelprobleem, ook het manoeuvreren door de straten zal ook een lastige taak zijn die opgelost dient te worden. Zo ontstaan bij ieder probleem weer andere problemen waar oplossingen voor gevonden moeten worden, geen taak zal echter hetzelfde zijn.

6.2. Situaties die de capaciteiten van mensen overschrijden

In sommige situaties moet er teveel informatie in te korte tijd verwerkt worden en daarom kan deze taak niet meer door mensen gedaan worden. Te denken valt aan situaties die te snel gaan voor de mens om op alle informatie die aangeboden wordt te reageren. Of het feit dat er zoveel gegevens aangeboden worden dat de mens die niet op tijd kan verwerken om nog de

goede beslissing te nemen. Tegenwoordig gaan de informatiestromen steeds sneller en de informatie wordt steeds gedetailleerder en meer. Men heeft al computerprogramma's nodig om al deze gegevens te verwerken, bijvoorbeeld door OLAP¹³ programma's. Op het moment dat op basis van deze gegevens beslissingen moeten worden genomen is de mens gewoon te laat.

Te denken valt aan een experiment dat door de NASA uitgevoerd is om op grote hoogte een vliegtuig op zijn eigen schokgolf te laten vliegen¹⁴. Er moet dan met een snelheid van mach 5 op atmosferische veranderingen geanticipeerd worden. De mens is daarvoor gewoon te traag. Nu al moeten piloten van gevechtsvliegtuigen bijgestaan worden door computers, die ervoor moeten zorgen dat het vliegtuig stabiel blijft. Natuurlijk moeten er dan sensoren zijn om de atmosferische veranderingen waar te nemen en voor het stabiel blijven van het vliegtuig zou ook een ARMA-model (zie § 5.3.3) gebruikt kunnen worden.

Ook de verwerking van alle informatiestromen in een fabriek is langzamerhand zo complex dat daar computers voor onmisbaar zijn. Men zou bijvoorbeeld een zelfadaptief systeem de voorraden kunnen laten beheersen. Vooral bij een JIT¹⁵-voorraadstrategie is de planning en het precies op tijd, aanleveren van de voorraden cruciaal. Om dat voor een hele fabriek bij te houden is voor een mens onmogelijk. Doordat een computersysteem dan steeds meer inzicht krijgt in de belevering (welke leveranciers leveren op tijd te vroeg te laat) kunnen de voorraadkosten geminimaliseerd worden.

Bij deze voorbeelden zien we dat de mens door zijn eigen vooruitgang ingehaald wordt. We weten zoveel te kennis in zo weinig tijd te produceren dat we computers nodig hebben om voor ons beslissingen te nemen, oftewel doordat een bepaald criterium wordt meegegeven aan de computer, oftewel door het op kleinere schaal aan de computer te leren en het dan op grotere schaal door deze computer uit te laten voeren. Omdat ook deze systemen kunnen leren van de fouten die ze in het verleden gemaakt hebben zullen ook deze systemen zelfadaptief zijn en dus steeds beter worden in hun taak.

6.3. De mogelijkheid om kennis te verspreiden.

Er zullen ook mogelijkheden zijn om door middel van zelfadaptieve systemen kennis te verspreiden. Iets wat op een bepaalde plek ontdekt is, een bepaalde systematiek een bepaalde manier om iets beter te doen kan zo verspreid worden. Het gaat juist weer om situaties die niet in regels te vatten zijn. Zo kan er ook op andere plaatsten geprofiteerd worden van de kennis die slechts bepaalde mensen hebben.

Zo rijden er in bepaalde delen van de Verenigde Staten mensen rond die rondkijken op zoek naar wervelstromen die zich gaan ontwikkelen. Daarbij kijken ze naar bepaalde patronen in de lucht, de temperatuur op dat moment en nog meer. Nu kan het zijn dat men daar een bepaald systeem ontwikkeld heeft dat geleerd heeft om deze mogelijke wervelstormen te ontdekken. Dit systeem zou dan ook in andere delen van de wereld gebruikt kunnen worden om mensen te waarschuwen.

Deze situatie is het lastigste om op korte termijn te realiseren, er moet namelijk eerst een goed werkend systeem op een bepaalde plek bestaan alvorens dat deze kennis verspreid en gedeeld kan worden. Toch kan het zeker op sommige gebieden zeer interessant zijn. Te denken valt dan aan de terreinen waarop er globale belangen bestaan, zoals in de medische wetenschap, of de hulpverlening aan mensen. Ook kan dit van belang zijn voor bedrijven om kennis die in het ene land toegepast wordt naar een andere plek te kunnen exporteren.

¹³ Online Analytical processing

¹⁴ [http://technology.nasa.gov/scripts/nls_ax.dll/w3SuccItem\(401240\)](http://technology.nasa.gov/scripts/nls_ax.dll/w3SuccItem(401240))

¹⁵ Just-in-time

6.4. De voorkoming van fouten

Ten vierde is er de mogelijkheid om zelfadaptieve systemen mee te laten draaien om fouten te voorkomen. Dit kan een oplossing zijn in drukke stressvolle situaties, maar ook in situaties waarbij de kosten van een fout zo hoog zijn dat dit eigenlijk niet voor mag komen. Het kan voorkomen dat mensen ondanks alle training en voorzorgsmaatregelen in paniek raken of de controles verliezen. Het mensen raken op het eind van de dag vermoeid en zijn dan niet meer helemaal scherp. Helaas kan dat dan tot verschrikkelijke gevolgen leiden. Om dat soort situaties te voorkomen zou men op de achtergrond zelfadaptieve systemen kunnen laten draaien, die de gang van zaken in de gaten houden en de mensen kunnen waarschuwen voor ongewenste situaties.

Hierbij valt te denken aan bijvoorbeeld de luchtverkeersleiding bij vliegvelden. Zoals de reclameslogan al zegt "it is an art"¹⁶ Vaak hoor je dat er een tekort aan deze mensen is en dat de mensen die er moeten werken vermoeid op hun werk zitten. Juist bij dit werk mogen fouten niet voorkomen. Maar bij een ongeluk werd ook naar de drukte bij de luchtverkeersleiding gewezen¹⁷. Blijkbaar is het luchtverkeersruim met conventionele technieken niet te overzien. Of een bepaalde controles op schepen, al de aanvaringen bij de Tricolor hebben uitgewezen dat ook daar niet alles vlekkeloos verloopt. Mensen die op de brug in slaap vallen of gewoon niet goed opletten. Als daar bepaalde extra maatregelen genomen zouden kunnen worden om de zeevaart veiliger te maken zou dat een groot aantal ongelukken per jaar schelen.

We zien hier dat de zelfadaptieve systemen echt een andere functie vervullen. Er is geen behoefte aan vervanging, daarvoor zijn de geschetste situaties te gevaarlijk om alleen aan computers over te laten. Maar toch als door het gebruik van dit soort systemen één ongeluk, één fout per jaar voorkomen kan worden dan zou het de investering al waard zijn.

6.5. Nieuwe situaties

Ten vijfde situaties waarin het niet mogelijk is en was om mensen te gebruiken. Door deze techniek kunnen nieuwe grenzen doorbroken worden. In deze situatie gaat het echt om het ontdekken van nieuwe mogelijkheden. Situaties die zonder deze techniek echt onmogelijk zouden blijven. Meestal zal deze taak niet eerder uitgevoerd zijn omdat de situatie voor mensen levensbedreigend is en waarschijnlijk zo erg dat het zou leiden tot de dood. En toch zouden we sommige verschijnselen graag nader willen onderzoeken. De mens wil alles weten en alles kunnen verklaren. Door deze systemen in te zetten zou het mogelijk zijn dat een robot bepaalde plekken of bepaalde verschijnselen te laten onderzoeken. Natuurlijk zal deze robot dan moeten weten wat belangrijk is wat wel kan en niet kan in bepaalde situaties. Dit zijn dan ook precies de dingen waarvoor een zelfadaptief systeem nodig is.

Te denken valt aan ruimtereizen of ontdekkingstochten naar gevaarlijke plekken. Daar komt men juist situaties tegen waarin men moet improviseren. Wat wel en wat niet te doen? Toch zijn daar wel bepaalde vuistregels voor te geven. Door middel van deze toepassing van zelfadaptieve systemen kan de mens uiteindelijk meer te weten kunnen komen over het heelal en over de aarde zelf, over hoe het was en wat we moeten doen om wat we hebben te behouden. Ook deze toepassing kan zeer nuttig zijn.

Dit zijn 5 situaties waarin een mogelijke toekomst zou kunnen liggen voor zelfadaptieve systemen. Misschien zijn er wel meer, dat zal de toekomst moeten leren. De weg naar het succes zal lang zijn en er zullen steeds kleine stappen gezet moeten worden. De barrières die geslecht moeten worden zullen echter niet alleen op technologisch gebied liggen ook de sociale aversie zal overwonnen moeten worden. Heeft de mens voldoende vertrouwen in de

¹⁶ reclamespot voor het verven van luchtverkeersleiding

¹⁷ Zie hiervoor schriftelijke vraag E-2130/02 en E-2159/02 aan de Europese commissie

techniek van zelfadaptieve systemen om deze techniek gemeengoed te laten worden? Is er voldoende sociaal draagvlak om deze technieken op grote schaal te introduceren? Allemaal vragen waarop het antwoord in toekomst ligt. Tot slot nog een ander voorbeeld van het gebruik van zelfadaptieve systemen, om aan te tonen dat er verschillende mogelijkheden zijn om deze techniek toe te passen.

6.6. ALVINN¹⁸

Tot slot wil ik uitgebreider op een andere toepassing ingaan, namelijk ALVINN (autonomous land vehicle in a neural network). In deze auto is een neurale netwerk ingebouwd dat de basis vormt voor een auto die zelfstandig kan rijden. Deze auto kan op wegen zonder hulp van mensen van A naar B komen. Het systeem werkt op de onderstaande manier.

Op de auto is een videocamera gemonteerd die de weg filmt. Dat wordt omgezet naar een 30x32 beeld en dat vormt de input voor het neurale netwerk. Dan is er een laag met 5 'hidden units'. Uiteindelijk is er een output die varieert van scherp naar rechts sturen tot rechtdoor, tot scherp naar links sturen. Zo zijn er 30 output units.

Deze techniek is op verschillende soorten weg getest en het is gebleken dat de auto op verschillende soorten wegen, modderige wegen eenbaans- en tweebaanswegen tot goede prestaties in staat bleek. Ook is gebleken dat deze techniek toegepast kan worden op burgervoertuigen. Het is dus mogelijk om straks een auto te hebben die zelfstandig van A naar B rijdt zonder dat de mens daar nog iets aan hoeft te doen. Op dit moment zal het voertuig nog aanpassingen nodig hebben om werkelijk altijd zelfstandig te kunnen rijden maar met iemand achter het stuur die in kon grijpen, heeft het voertuig al zelfstandig 90 km op de snelweg gereden. Hier moet wel bij vermeld worden dat het gaspedaal nog door de mens bediend werd, ALVINN is puur gemaakt om een voertuig op de juiste weg te houden, voor het aanpassen van de snelheid is aan de voorganger is een ander systeem vereist.

Een ander nadeel is dat het systeem rijdt op basis van verouderde beelden. Doordat de data eerst bewerkt moeten worden en dan aangeboden aan neurale netwerken is de informatie iets vertraagd waarop de auto zijn beslissingen neemt iets vertraagd. Echter met de huidige stand van zaken op het gebied van processoren mag dit verschil in tijd eigenlijk geen naam meer hebben.

Is dit een zelfadaptief systeem? Niet compleet want zoals al opgemerkt er moeten nog een groot aantal onderdelen bij om het echt een zelfadaptief systeem te laten zijn. Snelheid geven en remmen; een onderdeel dat bepaald wanneer er wel en niet ingehaald dient te worden; een onderdeel dat aangeeft dat er getankt moet gaan worden; een routeplanner. Kortom het is nog geen systeem dat alles zelf doet. Maar het begin is er. Als alle gegevens omtrent de auto in een database opgeslagen kunnen worden is het zeer goed mogelijk om op basis van die gegevens de auto verder te laten leren. Er moeten natuurlijk wel grenzen bepaald worden, wanneer mag deze auto nog wel de straat op. Als het glad is of het mist dan zal het een stuk lastiger zijn om de auto zelf te laten rijden. Het zal zeker niet onder alle mogelijke omstandigheden mogelijk zijn om met een auto te rijden, maar als die alleen al de lange stukken op de autosnelweg zou kunnen overnemen zou dat een revolutionaire ontwikkeling zijn.

¹⁸ zie http://www.ri.cmu.edu/projects/project_160.html



Fig. 3. The Carnegie Mellon NAVLAB autonomous navigation testbed.

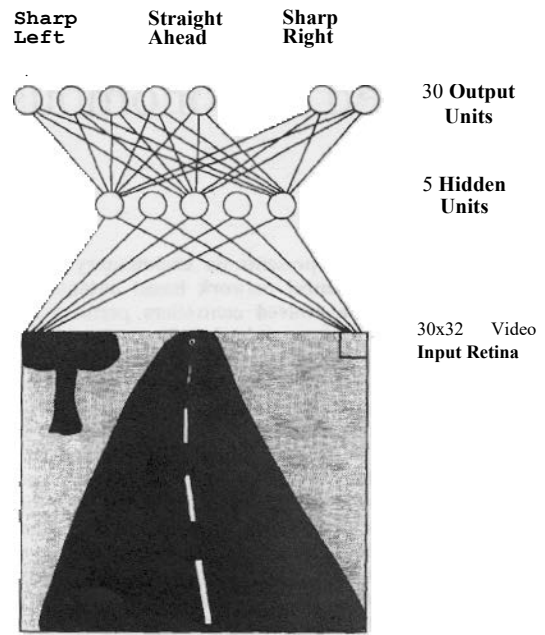


Fig. 4. The ALVINN neural network architecture

7. Overwegingen bij het ontwikkelen van een zelfadaptief systeem

In het vorige hoofdstuk is aangegeven in welke situaties het nuttig zou kunnen zijn om aan een zelfadaptief systeem te gaan ontwikkelen. Tot slot wil ik nog een aantal opmerkingen maken bij het ontwikkelen van een zelfadaptief systeem.

Ten eerste is het zo dat voor elk project, voor de ontwikkeling van elk product geldt dat tijd maal geld is resultaat. Hoe meer geld en tijd er in de ontwikkeling gestoken kan worden hoe beter het resultaat. Ook moet er een stuk committent zijn bij de opdrachtgever. Dit zijn allemaal zaken die terugkomen in het projectmanagement.

Met mijn eerste opmerking kom ik eigenlijk terug bij figuur 1. De afgelopen tijd hebben er door mijn hoofd tal van mogelijkheden gespeeld voor het toepassen van een systeem. Een grens ben ik echter wel tegengekomen, als er gecommuniceerd moet worden met mensen, op welke manier dan ook dan zal deze taak beter door mensen zelf gedaan kunnen worden. Een van de voorbeelden die ik in mijn hoofd had was de vervanging van bepaalde taken van een verpleegkundige om te helpen bij bijvoorbeeld het uitdelen van de medicijnen door een zelfadaptief systeem. Maar omdat daar zoveel andere neventaken bij zitten zoals een praatje maken het checken of het goed gaat met de patiënt, het controleren of de medicijnen daadwerkelijk ingenomen worden en nog veel meer, leek het mij onwenselijk om deze taak door een zelfadaptief systeem te laten vervullen. Het communiceren met mensen voldoet namelijk niet aan de voorwaarde dat de optimale oplossing voor iedereen bekend is en dat er 1 oplossing algemeen als de beste gezien wordt.

Een ander punt is of een taak op te splitsen is in meerdere zelfadaptieve systemen opgedeeld kan worden, zoals dat onder andere bij de syllogie gedaan is (zie ook Simmons). Het is namelijk eenvoudiger om regels te vinden voor kleinere deelproblemen dan voor het grotere probleem. Dat blijft te lastig. Het lukt niet om in een systeem alle taken uit te laten voeren, die een mens in zijn eentje kan doen aan boord van een zeilboot of achter het stuur van een auto. Juist door een goede hiërarchie en opdeling van deelproblemen te maken is het mogelijk dit soort problemen op te lossen.

Ook is het belangrijk om te blijven kijken en redeneren op de manier zoals ervaren mensen, de experts op het gebied, naar een bepaalde zaak kijken. Meestal is daar een heleboel informatie uit te halen. Waarom benoemen de experts bepaalde zaken op die manier. Wat zit daar achter. Als het een andere naam heeft zal het ongetwijfeld ook een andere functie hebben. Juist in de zeilwereld komt dat zeer goed naar voren. Elk lijn heeft zijn eigen naam, een val, een schoot, een talie, een strekker. Het zijn allemaal lijnen maar door de benaming is gelijk duidelijk wat de functie van deze lijn is. Zo zullen er parallellen in andere situaties. Probeer niet door deze terminologie tegengewerkt te worden, maar probeer deze juist in je voordeel te laten werken. Maak er gebruik van.

Naast dit alles speelt ook nog de maatschappelijke discussie. Vindt het publiek het verantwoord en acceptabel dat sommige taken straks door robots gedaan worden. Wil men wel rondgereden worden door een zelfrijdende auto of bestaat daar grote aversie tegen. Over dit probleem zal ook ontzettend moeten worden nagedacht als men een zelfadaptief systeem op grotere schaal in wil voeren. Vaak als men met de syllogie in de haven ligt dan krijgen de zeilers ook opmerkingen te horen als ga weg met je speeltje, zeilen is een mensensport die door mensen niet door computers gedaan moet worden. Anderzijds is dat ook het hoofd in het zand steken. Bij de laatste Volvo Ocean race waren er systemen van robosail aan boord van een van de schepen¹⁹. In de top gaat zoveel geld en om dat men er alles voor zal doen om de tegenstander vooruit te blijven. Dit is natuurlijk ook zo in het bedrijfsleven. Als men door deze ontwikkeling een beslissende voorsprong op de concurrentie kan nemen zal een bedrijf

¹⁹ Aan boord van de Djuice dragons

het zeker niet laten. En dan moet de rest ook volgen. Immers als er een schaap over de dam is...

Tot slot nog een opmerking. 'Een zelfadaptief systeem is een kind dat je alles moet leren' een opmerking van M. van Aartrijk van Robosail. Het is ook nog zo dat een zelfadaptief systeem geen gevoel heeft om dingen op te steken. Het systeem moet alles verteld worden en er moeten regelmatig fouten uitgehaald worden. Men kan een aantal keer de wereld rondzeilen maar als men zich niet intensief met het leeralgoritme bezighoudt dan is het de vraag of de algoritmen zeilen geleerd hebben of de stromingen van het water.

Cruciaal is natuurlijk het vinden van de juiste attributen die de waarde van de uitvoer bepalen. Bij het zeilen is natuurlijk de windrichting en snelheid belangrijk, bij autorijden de afstand tot de voorganger en wat de maximumsnelheid op dat weggedeelte is. Zo heeft iedere situatie zijn eigen kenmerken die bepalen welke actie er ondernomen moet worden. Voor het goed functioneren is het noodzakelijk deze kenmerken te onderscheiden. Het bepalen van de attributen kan natuurlijk op bekende klassieke manieren gebeuren als bijvoorbeeld door het bepalen van de correlatie tussen het attribuut en uitkomst van de doelfunctie, maar er zijn natuurlijk ook andere methoden denkbaar, bijvoorbeeld door het toepassen van datamining. Het gaat om het zoeken naar de verbanden tussen twee en misschien wel meerdere omstandigheden. Om hier inzicht in te krijgen zal er veel met experts gepraat moeten worden om te bepalen op basis waarvan zij de beslissingen nemen. En dan komt de moeilijkheid om deze omstandigheden werkelijk waar te nemen. Zijn deze ideeën ook werkelijk verifiëren, op basis van de data die verzameld zijn. Al met al zal dit een van de lastigere onderdelen zijn van het bouwen van een zelfadaptief systeem.

Tot slot de opmerking dat we aan computers nooit dingen kunnen leren die de mens zelf niet begrijpt. Dat geldt voor de problemen die de mens niet begrijpt zoals de effectenbeurs maar ook situaties waarin men terecht kan komen. De syllogie kwam bijvoorbeeld in een orkaan (ongeveer 80 knopen wind) terecht waarbij mens en computer maar een doel hadden en dat was overleven. Daar zit wel een bepaalde zwakte van dit soort systemen, onder extreme omstandigheden niets verwachten van een zelfadaptief systeem, daar is het systeem eenmaal niet voor getraind. In die situaties zullen het toch altijd mensen moeten zijn die de beslissingen nemen.

8. Conclusie

De vraagstelling waar ik aan het begin van dit werkstuk mee begon was:

Kan de techniek die ontwikkeld is om een zeilboot zelf te laten zeilen, toegepast worden in een bedrijfseconomische omgeving?

Een vraagstelling die zich niet in een stap liet beantwoorden. Na duidelijk gemaakt te hebben om welke techniek het hier gaat en mogelijke toepassingen te hebben onderkent, kom ik tot de slotsom dat er wel degelijk mogelijkheden zijn om deze techniek toe te passen, ook in bedrijfseconomische omgevingen.

In de komende tijd waarin er steeds meer data sneller verwerkt zullen moeten worden zal deze techniek een oplossing zijn om de daarbij komende problemen op te lossen. Ondanks dat deze techniek nog in zijn kinderschoenen staat en de mogelijkheden nog niet geheel duidelijk zijn. Is het belangrijk voor bedrijven en instellingen om de ontwikkelingen op het gebied van zelfadaptieve systemen te blijven volgen om te boot niet te missen. Immers snelle verwerking van data, en op basis van deze data computers beslissingen laten nemen is de wereld van de toekomst. De mens zal verder willen en steeds sneller en steeds meer. Grenzen vervagen in de moderne maatschappij. De mens ontwikkelt wel, maar in sommige opzichten langzamer dan de wereld om zich heen. De mens heeft 1 paar ogen en 1 paar oren, dat zullen er waarschijnlijk nooit twee worden. Twee paar ogen bij een robot is makkelijker te realiseren. Daardoor is er zeker een toekomst voor zelfadaptieve systemen op het gebied van fouten voorkomen, vervangen van mensen, kennis verspreiden, meer en sneller gegevens kunnen verwerken en het inzetten van robots in nieuwe situaties.

De ontwikkeling van de soft- en hardware aan boord van de syllogic is slechts een kleine stap op weg naar de vele mogelijkheden die er zijn op het gebied van zelfadaptieve systemen. Om het in termen van Deze ontwikkeling bevindt zich in zijn beginstadium. Het is om in termen van de portfolio matrix van de Boston Consulting Group te beschouwen het is nog een 'wild cat' en het kan zeker een star worden, maar dan moet er voldoende techniek en mogelijkheden ontwikkeld worden deze techniek van zelfadaptieve systemen tot wasdom te laten komen. Mogelijkheden voor het toepassen zijn er te over, maar het zal de kunst zijn om de situaties te onderkennen en een goede afweging te maken van de kosten en de baten. Een goede afweging daartussen zal ieder bedrijf voor zichzelf moeten maken. Maar een auto die compleet zelfstandig van A naar B zal kunnen rijden, daar zal zeker markt voor zijn. Net zo als er nu een markt is voor het maken van autopiloten en aanverwante producten aan boord van schepen. Het is de stap nemen, het eerste schaap zijn dat over de dam gaat.

Op dit moment wil ik geen voorspelling doen ten aanzien van de ontwikkeling van zelfadaptieve systemen. Daarvoor zijn de afgelopen decennia op informatica gebied de zaken te snel ontwikkeld maar ondanks alles konden ze niet altijd de hooggespannen verwachtingen waarmaken. De toekomst blijft onzeker, maar dat er steeds meer zaken sneller en efficiënter door computers gedaan zullen gaan worden, daar twijfel ik niet aan, de ontwikkeling van zelfadaptieve systemen is weer een kleine stap voorwaarts.

9. Litteratuurlijst

- [1] Van Aartrijk, Tagliola en Adriaans, AI on the ocean 2002
- [2] Baluja, Evolution of an artificial neural network based autonomous land vehicle controller, 1996
- [3] Heinemann, Convergence of adapting learning and the concept of expectational stability in linear rational expectation models with mutiple equilibria, 1997
- [4] Hoefnagels, Het Zeilboek, 1996
- [5] Mitchell, Machine learning, 1998
- [6] NASA succes story; [http://technology.nasa.gov/scripts/nls_ax.dll/w3SuccItem\(401240\)](http://technology.nasa.gov/scripts/nls_ax.dll/w3SuccItem(401240))
- [7] A brief introduction to modern time series
- [8] Pormerleau, Defense en Civilian applications of the ALVINN robot driving system, nov 1994
- [9] Simmons, Goodwin, Haigh, Koenig and O'Sullivan, A layered architecture for office delivery robots 1997
- [10] Tijms en Kalvenhagen, Modelbouw in de operations research, 1994