

# Leeralgoritmen, heuristieken en instructie-algoritmen

L. DE LEEUW

Vrije Universiteit, Amsterdam

## Samenvatting

*De literatuur op het gebied van het leren hanteren van algoritmen en heuristieken wordt kritisch beschouwd. Daarbij krijgt met name het werk van Landa aandacht. De door Landa niet aangeroerde of de door hem niet opgeloste belangrijke vragen op dit terrein worden behandeld.*

*Er wordt onderscheid gemaakt tussen leeralgoritmen en instructie-algoritmen, waarbij wordt betoogd dat aan het ontwikkelen van instructie-algoritmen in het algemeen te weinig aandacht wordt geschonken. Het standpunt wordt ingenomen dat ook als heuristische processen worden geleerd, werken volgens een instructie-algoritme wenselijk is.*

### 1. Landa over het aanleren van algoritmische processen; een onvolledige oriënteringsbasis

De pleidooien voor algoritmisering van handelingsvoorschriften en probleemoplossingsprocessen die Landa met veel verve in de westerse landen weet te houden, krijgen veel aandacht. En terecht! Een groot deel van de informatie die in het dagelijks leven tot ons komt is te vaag, te ingekleed, te weinig analytisch gesteld. Niet altijd is de voor de hand liggende, primaire reactie om deze informatie terzijde te schuiven een juiste en toegestane reactie. Vaak moet, ondanks een frustratie en ergernis over de slecht gestelde informatie, deze toch worden doorgeploegd, omdat aan verplichtingen moet worden voldaan (bijv. invullen van formulieren) of doelen anders niet bereikt worden (bijv. de weg zoeken in een stad, op de hoogte blijven op belangrijke gebieden). Het pleiten voor meer helderheid en duidelijkheid op dit terrein is dan ook een zinvolle zaak en Landa geldt hier als een goed advocaat die met duidelijk gekozen voorbeelden weet te overtuigen. Dat ook binnen de onderwijs-leersituatie de leerlingen op een aantal gebieden in het vage kunnen worden gelaten en dat alertheid op het punt van eenduidigheid van te nemen oplossingsstappen de prestatie kan verbeteren, daarvan

wordt men na lezing van Landa's publikaties (Landa, 1969, 1976) al snel overtuigd.

Toch laat de auteur ons met een groot aantal vraagpunten zitten. Die vraagpunten zijn niet gelegen op het gebied van het vaststellen van het algoritme dat *in principe* tot de oplossing van een probleemcategorie leidt. Een gedegen logische analyse kan leiden tot de vaststelling van een sluitend algoritme; indien een dergelijk algoritme correct wordt doorlopen, de juiste operaties en beslissingen worden uitgevoerd, is er sprake van oplossingsgarantie. De vraag is echter of het algoritme dat door middel van logische analyse tot stand is gekomen het algoritme is zoals dat aan de leerling moet worden gepresenteerd. Een vraag die daaraan vooraf gaat is of het algoritme in kant-en-klare vorm moet worden aangeboden of dat dit (mede) door de leerling moet worden afgeleid.

Een volgend heel belangrijk vraagpunt is hoe algoritmen moeten worden aangeleerd. Daarbij is allereerst de vraag van belang hoe op grond van het gegeven (visuele of verbale) algoritme een algoritmisch proces tot stand kan worden gebracht. Een veelal verwaarloosd punt is hoe dat algoritmische proces op gang te houden nadat de hulp van buiten (het visueel of verbaal gepresenteerde algoritmische schema) is weggenomen. Het gaat m.a.w. om de vraag hoe de leerling tenslotte over een geïnternaliseerd, geautomatiseerd, eventueel verkort algoritme kan komen te beschikken. We kunnen deze vraagpunten als volgt kort weergeven.

1. Moet het algoritme in kant en klare vorm worden aangeboden of moet dit geleidelijk aan, samen met de leerlingen worden opgebouwd? Verwante vragen zijn: a) Moet het algoritme uitsluitend zeer uitvoerig worden aangeboden en moet verkorting hiervan worden overgelaten aan de leerling, of moet ook de verkorting en automatisering worden begeleid (progressieve schematisering)? b) Moet het algoritme worden aangeboden (opgebouwd) zoals dat door een ervaren deskundige wordt gehanteerd (expert-strategie) of kan beter minder 'sophisticated' worden begonnen (novice-strategie)?

2. In welke vorm moet het algoritme worden gepresenteerd en geoefend teneinde zo snel mogelijk, met zo min mogelijk fouten en inspanning en zo lang mogelijk (retentie) beheerst te worden?
3. Volgens welke leerprincipes kan een geleidelijke terugneming ('vanishing') van het aanvankelijk extern gegeven algoritme tot stand worden gebracht, zodanig dat de leerling tenslotte zonder visuele of verbale steun tot oplossing van de betreffende klasse van problemen in staat is?
4. Geldt voor de beantwoording van de bovengenoemde vragen één uniforme regel of dient het antwoord te worden gegeven gedifferentieerd naar bepaalde typen leerlingen (persoonskenmerken)?

Landa stelt in zijn eerste hoofdwerk (Landa, 1969) dat het niet verstandig is de algoritmen in kant en klare vorm aan te bieden. De leerling moet het algoritme òf zelf *afleiden* òf althans *begrijpen* waarom het algoritme deze vorm heeft. Het lijkt op het eerste gezicht aanbevelenswaard om te stellen dat *elk* algoritme door de leerling dient te worden afgeleid (d.m.v. zelf-ontdekking of geleide ontdekking). De vraag is echter of elke leerling daartoe voor elk type algoritme in staat is. Ook Boekaerts, (1978, blz. 268) stelt dat 'it would be time consuming to teach every new rule or plan by the self-discovery method'. Bovendien geldt volgens haar dat er algoritmen zijn die zo ingewikkeld of moeilijk zijn dat ze niet door leerlingen ontdekt of afgeleid zouden kunnen worden. Te denken valt aan regels voor het worteltrekken of natuurkundige bewijzen. Maar ook ingeval de leerkracht het algoritme presenteert is het aan te bevelen vooraf tijd te besteden aan een analyse van het probleem teneinde de leerlingen vertrouwd te maken met het probleem en een probleem-set te creëren, zodat de leerlingen gemotiveerd worden om het door de leerkracht verschaft plan van handelen te leren (Boekaerts, 1978, blz. 274). Uit de hiervoor genoemde visies op het probleem: geven vs. zelfstandig afleiden van het algoritme, wordt duidelijk dat het in elk geval van belang is, dat de algoritmen niet op een 'domme' wijze aan de leerlingen worden gepresenteerd, maar dat de leerling het waarom ervan begrijpt. In eigen onderzoek (De Leeuw, 1979) hebben we de aan te leren algoritmen samen met de leerling, stap voor stap ontwikkeld, zodat de leerling de ratio van de algoritmen begrijpt<sup>1</sup>.

## 2. Andere auteurs over het aanleren van algoritmische processen

Het blijkt onjuist om te veronderstellen dat een eenmaal aangeleerd algoritmisch proces na verloop van tijd in exact dezelfde vorm zal worden voltrokken. Mensen komen tot wijzigingen, verkortingen, efficiëntere uitvoering van dit proces. Er is m.a.w. sprake van ideosyncratische algoritmen.

Resnick (1976) heeft beschreven hoe leerlingen eenmaal geleerde algoritmen later op hun eigen wijze gebruiken. Bij het leren optellen en aftrekken door kinderen in de kleuterleeftijd en in de lagere klassen van de basisschool bleek dat de leerlingen in een later stadium van het leerproces de aangeleerde algoritmen transformeren tot een voor hen meer efficiënte werkwijze. Uit de experimenten bleek dat 'children seek simplifying procedures that lead them to construct or "invent" more efficient routines that might be quite difficult to teach them directly'. Resnick maakt op grond hiervan onderscheid tussen *rationele* en *empirische* taakanalyse. Een rationele taakanalyse wordt gedefinieerd als een poging om processen en procedures te specificeren zoals die afgeleid kunnen worden uit theoretische denkbeelden over efficiënt taakgedrag. Bij een rationele taakanalyse die tot dit soort 'geïdealiseerde' prestatie leidt worden weinig expliciete assumpties gemaakt over beperkingen van de menselijke geheugencapaciteit of van de perceptuele coderingsprocessen. Een empirische taakanalyse daarentegen gaat uit van een zeer competente 'skilled performer'. De vraag is nu wat in het onderwijs te doen. Moet worden afgestapt van het aanleren van algoritmen die gebaseerd zijn op de rationele taakanalyse en de meer empirisch gebaseerde taakanalyse als uitgangspunt worden genomen? Moeten we m.a.w. besluiten dat de instructie niet moet uitgaan van de rationele analyse omdat deze niet met de 'skilled performance' overeenstemt? Resnick is van mening dat dit onjuist zou zijn. Een dergelijk uitgangspunt zou berusten op de onjuiste assumptie dat efficiënte instructie *noodzakelijkerwijs* instructie in 'skilled performance' strategieën zou zijn, in plaats van instructie die de leerling in een goede positie brengt om zelf efficiënte strategieën te ontdekken, uit te vinden. Want dat is wat de leerlingen in de betreffende onderzoeken bleken te doen. Ze leerden een routine maar vonden toen zelf een meer efficiënte werkwijze uit. Resnick vat dit als volgt kernachtig samen: 'People apparently invent even within the confines of algorithmic instruction'. Mensen – zelfs jonge kinderen – gebruiken feedback uit hun omgeving om hun routines te vereenvoudigen. Ze accepteren de hen aangeleerde routines niet als 'gegeven',

maar als uitgangspunten. Ze blijven inventief.

Deze visie is van belang voor het leren op school. Het onderwijsproces is immers niet altijd optimaal, de gekozen voorbeelden zijn niet altijd elegant en toch wordt verwacht dat de leerlingen de onderliggende principes zelf ontdekken. Dit suggereert dat verschillen in leervermogen – vaak aangeduid als intelligentie of 'aptitude' – in feite verschillen kunnen zijn in de hoeveelheid steun die leerlingen nodig hebben om via de nodige inventiviteit de graad van vereenvoudiging en georganiseerdheid te bereiken die kenmerkend is voor de prestatie van een 'skilled performer' op het gebied. Sommigen zoeken en vinden orde in de meest ongeordend gepresenteerde leerstof, de meesten doen het goed als de instructie de onderliggende structuur goed laat zien en weer anderen hebben expliciete hulp nodig om de efficiënte strategieën te vinden. Mensen verschillen dus sterk in inventiviteit. 'Leren leren' dient, aldus Resnick, dan ook neer te komen op het aanleren van inventie- en ontdekkingsstrategieën die de leerling minder afhankelijk maken van de onderwijslegantie van de leerkracht.

Ook Gal'perin (1978) is overigens van mening dat de in de volledige oriënteringsbasis aangeboden werkwijze (vaak een algoritme) niet letterlijk wordt voltrokken door de leerling zoals gepresenteerd. Men moet, aldus Gal'perin, het *schema* van de volledige oriënteringsbasis (Ohv) niet verwarren met de *eigenlijke* oriënteringsbasis. Het *schema* van de Ohv is een uitwendig gegeven model; de *eigenlijke* Ohv echter is een 'weerspiegeling' van dit model in de psyche van de leerling. Het model blijft natuurlijk constant, maar even natuurlijk wijzigt zich de weerspiegeling ervan (de reële en actieve Oh) in de psyche van de leerling. Deze veranderingen in de actieve Ohv komen in de verschillende, door Gal'perin onderscheiden, parameters naar voren en worden bepaald door de mate van 'inoefening' van de handeling op verschillende niveaus. Met de zich wijzigende oriëntering verandert ook de kwaliteit van de handeling.

### 3. Het aanleren van algoritmische processen; de noodzaak van instructie-algoritmen

In een eerdere publikatie in dit tijdschrift (De Leeuw, 1975) hebben wij een aantal door Landa in zijn eerste hoofdwerk (1969) genoemde stellingnamen m.b.t. het aanleren van algoritmische processen vermeld. Ook werden daarbij kritische kanttekeningen gemaakt bij de onderwijsprocedure-beschrijving van het aanleren van algoritmische processen op grammatica-gebied. Met name het gebrek aan eenduidig-

heid van de reactie-voorschriften van de onderwijzende instantie (wat te doen bij welke leerling-actie of -reactie) was daarbij punt van kritiek. In Landa's latere werk (1976) wordt een hoofdstuk gewijd aan 'Algorithms and programmed instruction: the theory and methodology of programming'. Een concrete werkwijze voor het aanleren van algoritmische processen wordt ook hier niet gegeven. Wel wordt een aantal voorschriften gegeven in nogal algemene termen. Zo wordt terecht gesteld, dat effectieve controle van een proces kennis vereist van de structuur van dat proces, d.w.z. dat een algoritmische beschrijving van het proces en het ontwerpen van daarmee corresponderende instructies die het proces controleren nodig is. Ook wordt hier het onderscheid tussen 'teaching (instructional) algorithms and the teaching of algorithms' door hem ingevoerd. We zullen in het vervolg resp. van instructie-algoritmen en leeralgoritmen spreken. Over hoe het instructie-algoritme moet worden afgestemd op het tot stand te brengen algoritmische proces, m.a.w. hoe het algoritmisch handelen gaandeweg moet worden aangeleerd, blijft veel in het vage. Wel bespreekt de auteur in het betreffende hoofdstuk de verschillende fasen in het schrijven van een instructie-programma om algoritmen aan te leren. Met name bij de beschrijving van de latere fasen die handelen over het vaststellen van de volgorde van de programma-onderdelen, het bepalen van het type leerling-activiteit die de overgang van de ene kennis-toestand naar de andere moet verzekeren en het programmeren van de feedback aan de leerling, zou men gerichte handelingsvoorschriften of -alternatieven verwachten. De aanwijzingen van Landa komen echter niet verder dan de vermelding van een aantal voorwaarden waaraan voldaan moet zijn. De in dat verband vermelde punten en principes zijn correct, maar zijn niet zodanig specifiek dat ze een éénduidige oplossing van het probleem, hoe een algoritmische procedure aan te leren, garanderen.

De door Landa aangereikte oplossingsmethode voor dit probleem is zeker niet algoritmisch van aard. Er worden uitsluitend algemene heuristische aanwijzingen gegeven. Zo wordt, als het gaat over het belangrijke punt hoe de volgorde van programma-onderdelen te bepalen teneinde tot beheersing van een algoritme te komen, gesteld dat een strikt logische analyse van het materiaal niet voldoende is, maar dat rekening moet worden gehouden met de wijze van assimilatie van het materiaal door de leerling, d.w.z. met de manier waarop deze het materiaal waarneemt en opslaat in zijn geheugen. Daarmee wordt niets gezegd over het belangrijke probleem hoe het algoritme geïntroduceerd moet worden, of alle stappen tegelijk moeten worden geoefend of achtereenvol-

gens apart, elk tot een beheersingskriterium (holistisch of serialistisch); ook de mate van 'overlearning' die gewenst is i.v.m. het vergeten komt niet aan de orde.

Over de hierna volgende fase, die betrekking heeft op het vaststellen van het type leerling-activiteit dat nodig is om elke volgende stap te leren beheersen, wordt gezegd dat kennis en vaardigheid niet in volledige vorm in het hoofd van de leerling kunnen worden geplant, maar dat de leerling deze moet assimileren door de bijbehorende operaties (cognitief of fysiek) zelf actief uit te voeren. De auteur verwijst hier, evenals in zijn vroegere werk, naar Gal'perin. Over het hoe hiervan wordt weinig specifiek gezegd.

Dat geldt nog sterker als het gaat om de keuze van de oefeningen door middel waarvan de gewenste vaardigheden moeten worden geleerd. Alleen het belang van de juiste keuze wordt benadrukt. Geen antwoord wordt gegeven op vragen als: moet het algoritmisch proces worden aangeleerd door het gebruik van een boom-algoritme of is een visueel schema een geschikt hulpmiddel; hoe moet de gegeven visuele steun geleidelijk worden teruggenomen ('vanished'), teneinde tot een zelfstandig oplossen (m.b.v. een geïnterioriseerd algoritme) te komen. Deze vragen worden afgedaan met: 'A detailed discussion of the question of how to select and devise instructional influences which meet these requirements would go beyond the scope of this chapter'.

Het is een niet te boude conclusie dat de verwijten die Landa herhaaldelijk maakt over het ontbreken van algoritmische voorschriften op een aantal kennisgebieden, in zijn richting gemaakt kunnen worden als het gaat om algoritmische voorschriften voor het aanleren van algoritmische processen. Anders gesteld: de *instructie-algoritmen* voor het inoefenen van leeralgoritmen ontbreken in Landa's werk in grote mate.

Om ervan verzekerd te zijn dat het algoritmisch proces door de leerling daadwerkelijk wordt ingeoeffend is een instructie-algoritme nodig waarin zijn vastgelegd: 1. de volgorde van de in te oefenen stappen; 2. bij welke leerlingactie met welke feedback en/of aanwijzing te reageren; 3. de criteria voor beheersing van de subalgoritmen, d.w.z. na welk beheersingsniveau de leerling het volgende subalgoritme of de volgende algoritme-stap mag gaan inoefenen; 4. in welk stadium de schema-steun wordt weggenomen en op grond van welke condities dit schema opnieuw wordt gepresenteerd.

Een pleidooi voor het ontwikkelen van instructie-algoritmen impliceert niet dat de algoritmische processen volgens een starre, uniforme werkwijze moeten worden aangeleerd. Het gaat erom dat door on-

derzoekers niet volstaan moet worden met het benadrukken van het belang van algoritmen-onderwijs, maar dat er onderzoek gedaan moet worden naar de meest optimale vorm waarin algoritmen kunnen worden aangeleerd. In een instructie-algoritme dat deze optimale onderwijsprocedure weergeeft zal overigens ruimte moeten worden gelaten voor individuele leerstijlen van leerlingen. We komen hierop nog terug.

#### 4. *Algoritmen versus visuele schema's; kritische kanttekeningen bij een tweetal studies*

Het ontbreken van expliciete voorschriften hoe het algoritmische proces in te oefenen speelt overigens in meer onderzoekingen een rol. In het door van Parrenen en Carpay (1972) beschreven onderzoek van Kuljutkin en Suchobskaja werd het effect vergeleken van twee psychologisch verschillende aanbiedingswijzen van wat logisch beschouwd één en hetzelfde algoritme was.

Van een onderwerp uit de schoolgrammatica werd aan één groep leerlingen dit algoritme gegeven in de vorm van een *'logische boom'*. De andere groep kreeg een *aanschouwelijk schema*. Daarnaast was er een controlegroep. Aan alle groepen werd tevoren een uitgebreide uitleg van de regels gegeven. Daarna werd de logische boom resp. het aanschouwelijke schema gegeven. De foutenpercentages waren voor beide vormen van visuele steun ongeveer gelijk. Het percentage fouten in de controlegroep was beduidend hoger.

Uit individuele experimenten kwam naar voren dat geen enkele proefpersoon, noch in de voortoets, noch na het onderwijs, bleek te werken met een getrapte beslissingsprocedure zoals die in de logische boom wordt gegeven. De beslissing verliep veelal *simultaan* en niet successief; er ontstond dus geen algoritmisch proces. Zo bleek met name bij het 'gebruik' van de logische boom, dat proefpersonen nog vaak intuïtief werken, niet systematisch successief nagaan welke kenmerken in de situatie aanwezig zijn. De conclusie van de onderzoekers is dat de uitvoerige vorm van handelen alleen in de mondelinge uiteenzetting plaats moet vinden. Wil men tot automatisch functionerende identificatie- en transformatie-vaardigheden komen, dan moet men de uitvoerige vorm van handelen niet laten 'stollen' tot een boom-model, maar verdient de afbeelding in de vorm van een aanschouwelijk schema de voorkeur. Het boom-model belemmert de verkorting van de handeling.

We willen bij deze conclusie een aantal kritische kanttekeningen plaatsen. Allereerst moet worden

opgemerkt, dat in dit experiment de handelingsstructuur niet werd geleerd *aan de hand van* het boomalgoritme of het aanschouwelijke schema. Het boomalgoritme werd gegeven *nadat* de regels uitvoerig aan de hand van voorbeelden waren behandeld. Er werd niet geoefend met de logische boom of het visuele schema naast de opgaven. Er werd m.a.w. geen algoritmisch proces ingeoeffend. De geneigdheid van leerlingen om niet systematisch te werken en stappen over te slaan werd hierdoor niet onderdrukt. Een goede onderwijsprocedure waarbij van algoritmische schema's gebruik wordt gemaakt dient de benodigde successieve werkwijze te bewerkstelligen. Pas als de leerling tot een successieve benadering in staat is (middels oefenen aan de hand van een algoritmisch schema, waarbij een geleidelijk terugnemen van de visuele steun moet plaatsvinden), dient tot verkorting van de handeling te worden overgegaan. Leidt de verkorte (evt. simultane) handeling op een zeker moment niet tot resultaat, maar tot een impasse dan moet de leerling kunnen terugvallen op de successieve werkwijze. Deze successieve werkwijze werd door Kuljutkin en Suchobskaja in het geheel niet getraind. De handelingsvoorschriften voor het inoefenen van het algoritmische proces ontbreken in dit onderzoek ten enen male.

Ook in het onderzoek van Wolters (1974) ontbrak een dergelijk instructie-algoritme. Zij onderzocht de vraag of met een anderssoortige algoritme dan de door Landa geschetste logische structuur-algoritmen niet een even goed of beter resultaat te bereiken zal zijn. Een logische structuur-algoritme (kortweg structuur-algoritme) is objectief van aard, in deze zin dat de juiste handelingen eenduidig en logisch correct worden voorgeschreven zonder daarbij echter rekening te houden met de werkbaarheid en leesbaarheid van dit algoritme. Als met deze factoren wel rekening wordt gehouden dan heeft het algoritme een subjectief karakter. Een algoritme waarin met de psychologische realisering van het rationele objectieve schema wordt rekening gehouden noemt Wolters een psychologisch werkalgoritme (kortweg: werkalgoritme). Zo'n werkalgoritme is subjectief van structuur. Een belangrijk verschil tussen een structuur-algoritme en een werkalgoritme is dat in het werkalgoritme overeenkomstige beslissingen in de verschillende takken voorkomen. Het is dus niet op een logisch zo economisch mogelijke wijze opgebouwd, zoals dat bij het structuur-algoritme het geval is.

Kuljutkin en Suchobskaja vergeleken het effect van een structuur-algoritme en een visueel schema. Wolters vergeleek het effect van een structuur-algoritme, een werkalgoritme en een aanschouwelijk

schema. Ze gebruikte hetzelfde soort materiaal als de Russische onderzoekers. Ze vond na twee trainingssessies de volgende resultaten. Er is geen significant verschil tussen de aanschouwelijke schemagroep en de werkalgoritme-groep. De aanschouwelijke schema-groep doet het significant beter dan de structuur-algoritme groep. Het verschil tussen de structuur-algoritme groep en de controle groep is niet significant. Bovendien geldt dat de aanschouwelijke schemagroep significant betere resultaten haalt dan de controle groep. Wolters concludeerde op grond van deze resultaten dat de interpretatie van Kuljutkin en Suchobskaja van de door hen gevonden verschillen onjuist is. Volgens deze auteurs is het algoritme minder effectief omdat het stapsgewijze karakter van het algoritme van negatieve invloed is op de resultaten. Omdat Wolters geen significant verschil in prestatie bij de resp. condities vond, wordt deze interpretatie op losse schroeven gezet.

Wij willen hier opnieuw een kritische kanttekening maken bij dit type experiment. Evenals bij Kuljutkin en Suchobskaja werden de proefpersonen van Wolters niet werkelijk getraind in het gebruik van de schema's. Na een viertal voorbeelden moesten haar proefpersonen met gebruikmaking van de geboden hulpmiddelen een twaalfstal opgaven oplossen. Ook van Parreren en Carpay (1972 blz. 133) zijn van mening dat het aanbieden van een algoritmisch voorschrift (boom-model) nog niet hoeft te leiden tot een algoritmisch proces. Van systematische training met feedback over de foute stappen (controle) was dus geen sprake. Bovendien bestond de natest niet uit opgaven waarbij de schema-steun was weggenomen. Er werd door de leerlingen dus niet aan de hand van een geïnterioriseerd schema gewerkt, maar m.b.v. een extern schema.

De vraag is hoe de verschillen zouden hebben gelegen als in de latere opgaven de schema-steun was weggenomen. Met gebruikmaking van de gegevens van Wolters is hierover een tentatieve uitspraak te doen. De proefpersonen moesten in de eindfase van het experiment het gebruikte schema *reproduceren*. Nu bleek de reproductie het hoogst te zijn voor de werkalgoritme groep, gevolgd door de aanschouwelijke schema groep en de structuur-algoritme groep.

De werkalgoritme groep bleek het schema dus het meest paraat te hebben. Dit maakt het waarschijnlijk dat als opgaven zouden zijn gepresenteerd *zonder schema-steun* de resultaten voor de werkalgoritme groep het hoogst zouden zijn geweest. Dit zou een conclusie zijn die volledig tegengesteld aan de bevindingen van Kuljutkin en Suchobskaja is.

Het is overigens denkbaar dat de vraag naar het differentieel effect van algoritmen (die tot een succesieve werkwijze leiden) en visuele schema's (simultane werkwijzen oproepend) niet in zijn algemeenheid beantwoord kan worden. Op grond van het werk van Pask en Scott (1971, 1972) kan onderscheid gemaakt worden tussen serialisten en holisten. Deze zijn resp. gekenmerkt door het successief en simultaan verwerken van informatie. Beishuizen (1975) omschrijft op grond van het werk van Pask en Scott de typen als volgt: *Holisten* leren, onthouden en recapitulieren informatie als een geheel: formeel, in termen van 'hoger orde' relaties. Ze vormen een afbeelding ('image') van een geheel systeem van feiten en principes. *Serialisten* leren, onthouden en recapitulieren informatie in termen van ketting-achtige cognitieve structuren, waarin items aan elkaar worden verbonden door simpele 'data-links': formeel, door 'lager-orde' relaties. Het ligt voor de hand om te veronderstellen dat serialisten meer baat zullen vinden bij een (werk)algoritme en holisten meer bij een visueel schema, omdat daarmee resp. successief en simultaan handelen wordt opgeroepen. Boekaerts (1978) brengt een aantal persoonskenmerken met gelijkgerichte predictie naar de informatie verwerving- en verwerkingskant (o.a. sharpening-leveling, reflectiviteit/impulsiviteit, veldonafhankelijk-veldafhankelijk) onder één noemer. Ze spreekt van *verbalizers-visualizers* die resp. gekenmerkt zouden zijn door het *sequentieel* en *parallel* verwerken van informatie. Deze dimensie lijkt gerelateerd te zijn aan de holist-serialist dimensie. Onderzoek naar de relevantie van de verbalizer-visualizer 'aptitude' in verband met een algoritmische versus een visueel schema 'treatment' lijkt ons zowel theoretisch als praktisch het entameren waard.

##### 5. Het aanleren van heuristische processen; keuze binnen een dilemma

Algoritmen zijn gekenmerkt door oplossingsgarantie. Dit impliceert dat de aanwijzingen binnen het algoritme zeer *specifiek* moeten zijn; zowel de condities voor het handelen als de uit te voeren operaties dienen nauwkeurig omschreven te zijn. Dit is niet het geval als het om heuristische regels gaat. Heuristische regels geven geen specifieke aanwijzingen voor het handelen. Ze plaatsen de oplosser slechts op het juiste denkspoor; de baan waarop de denker wordt geplaatst is tamelijk breed. Weliswaar zijn er binnen de categorie van heuristische regels of -oplossingsmethoden graden van specificiteit te onderscheiden (de aanwijzingen variëren van zeer vage tot

meer specifieke) maar de heuristische aanwijzingen zullen niet zo specifiek zijn, dat ze zonder meer tot de oplossing leiden. In dat geval zou immers geen heuristisch proces ontstaan: een heuristisch proces is gekenmerkt door eigenschappen als *zelfbepaaldheid* en *onafhankelijkheid* en daar is in geval de aanwijzingen (cues, hints) zonder meer tot de oplossing leiden geen sprake van.

De tegenstelling tussen *specificiteit* van aanwijzingen (oplossingsvoorschrift) en *zelf-bepaaldheid* van de acties (oplossingsproces) vormt het belangrijkste *dilemma* waarvoor men komt te staan als heuristische processen dienen te worden aangeleerd. Het (uiteindelijke) doel van instructie-programma's op dit gebied dient te zijn de leerling tot zelf-bepaalde heuristische processen te brengen.

Wil het denkproces op gang worden gebracht of gehouden teneinde (nieuwe) potentiële explicatieve relaties te zoeken, dan zijn zeer vage, algemene aanwijzingen onvoldoende, terwijl te specifieke aanwijzingen niets aan de inventiviteit van de leerling overlaten.

In ons onderzoek (De Leeuw, 1979) hebben we ter oplossing van dit probleem als uitgangspunt genomen dat binnen het instructie-programma niet gekozen hoeft te worden voor één vaste graad van specificiteit van de aanwijzingen, maar dat de specificiteit *cumulatief* kan toenemen als minder specifieke aanwijzingen ineffectief blijken.

Om een dergelijke, tamelijk subtiele, sturing van het oplossingsgedrag te realiseren dient voorafgaand aan de constructie van het instructie-programma goed nagedacht te worden over mogelijke oplossingspogingen (verstandige en minder verstandige, inclusief vergissingen) van de leerling. Bij elk van deze oplossingspogingen dient adequate feedback gereed te worden gemaakt. De heuristische hulp die – op verzoek van de leerling of op grond van zijn leergeschiedenis – wordt gegeven dient te worden geconstrueerd. Het instructie-plan dat op die manier ontstaat dient in hoge mate *specifiek* te zijn. De instructie kan volgens dit plan worden gepresenteerd door een leerkracht, maar kan ook in de vorm van geprogrammeerde instructie of computer-gestuurd onderwijs – waarbij de leergeschiedenis van de leerlingen kan worden opgeslagen en verwerkt – worden gepresenteerd. De vraag of het mogelijk is om met behulp van geprogrammeerde instructie de operaties die voor het heuristische proces nodig zijn te ontwikkelen, wordt ook door Landa (1976) gesteld. Teneinde het verloop van de creatieve zoekprocessen met behulp van geprogrammeerde instructie te ontwikkelen, d.w.z. op een stap-voor-stap manier met behulp van feedback, is het noodzakelijk:

1. de handelingen (responsen) van de leerling te voorspellen, zodat de daarbij behorende feedback kan worden voorbereid. Volgens Landa geldt dat 'this question is extremely complex and has not as yet been fully resolved'. De leerstof-programmeur moet de mogelijke wegen waarlangs de oplossing kan worden gezocht kennen en een aantal mogelijke acties voorzien. Maar een heuristische werkwijze blijft een heuristische werkwijze en daardoor zijn niet alle leerling-acties te voorzien. Daarom moet er een restcategorie van niet-voorzien antwoorden bestaan waarbij op de een of andere wijze aansluitende feedback gereed moet worden gemaakt.
2. De efficiënte stapgrootte voor het sturen en controleren van dergelijke processen vast te stellen. Het lijkt misschien twijfelachtig of het mogelijk is om heuristische zoekprocessen volgens een stap-voor-stap methode te ontwikkelen. Heuristische processen waaraan zelforganisatie en onafhankelijkheid inherent zijn impliceren immers non-determinering of althans onvolledige determinering van het gedrag door de instructie. Toch is hier geen sprake van een contradictie. Stap-voor-stap ontwikkeling en controle van heuristische processen is immers niet identiek met operatie-voor-operatie ontwikkeling en controle daarvan. De stapgrootte is variabel van aard en dient een functie te zijn van de onderwijssituatie. Landa wordt erg vaag als het gaat om een nadere precisering van de stapgrootte. 'What this step size should be for the development of creative processes is a special problem which goes beyond the issues under consideration at this particular point'.

#### 6. Het aanleren van heuristische processen; de noodzaak van een instructie-algoritme

Het probleem dat Landa hier aan de orde stelt heeft in feite betrekking op het ontwerpen van een *instructie-algoritme*. We spreken met opzet van een *instructie-algoritme*, ook als het om het aanleren van heuristische processen gaat. De reactie van de onderwijzende instantie mag niet van goede invallen afhankelijk zijn; het denkwerk over de meest optimale vorm van richting geven aan het denkproces dient voorafgaand aan het presenteren van de leer(denk)stof plaatsgehad te hebben. Het moet duidelijk zijn dat het hanteren van een instructie-algoritme niet hoeft te leiden tot een algoritmisch denkproces bij de leerling. Ook als gepoogd wordt de leerling inventief te laten bezig zijn, zelf-bepaaldheid

van zijn acties wordt nagestreefd, dan kan op grond van registratie van zijn acties – acties die van hem worden verlangd in de vorm van tussenstappen in het oplossingsproces – eenduidig bepaald worden welke hulp moet worden geboden. Deze eenduidigheid – bij bepaalde condities bepaalde instructie-operaties – maakt dat we van een instructie-algoritme moeten spreken.

De stapgrootte waar bij Landa sprake van is (zie hiervoor) hangt naar onze mening samen met de specificiteit van de te geven aanwijzingen. Als men vilstaan met zeer algemene aanwijzingen (bijv. 'probeer het eens op een andere manier') dan zal de stapgrootte aanzienlijk zijn. Als specifiekere aanwijzingen worden gegeven ('je kunt in plaats van optellen hier ook vermenigvuldigen'; 'je kunt de cirkels ook nog overlappend tekenen', e.d.) dan is de stapgrootte geringer. We hebben in ons onderzoek (De Leeuw, 1979) gekozen voor een *variabele* mate van specificiteit van de aanwijzingen. Deze benadering houdt in dat bij het eerste vastlopen van de leerling zeer algemene hulp geboden wordt; deze hulp wordt als ze te algemeen blijkt om tot een oplossing (spoging) te leiden gaandeweg specifieker. Deze werkwijze komt overeen met het door Selz (1935) genoemde principe van de 'kleinstmögliche Hilfe'. Er wordt dan hulp geboden op verschillende niveaus; hulp op het meer specifieke niveau wordt alleen dan geboden als de minder specifieke hulp niet effectief blijkt. De inhoud van de geboden hulp dient idealiter aan te sluiten bij de leergeschiedenis van de betreffende leerling; hoe vaak eerder binnen het probleem om hulp gevraagd, aansluiting bij die reeds ondernomen oplossingspoging die het meest in de richting van het juiste antwoord leidt, e.d. De inhoud van de te bieden hulp is dan dus afhankelijk van een aantal condities. Op grond van deze condities dient met bepaalde instructie-operaties gereageerd te worden. Deze condities en operaties zijn vastgelegd in het instructie-algoritme.

#### Noten

1. Een soort tussenvorm binnen de hier geschetste problematiek is om leerlingen tamelijk ingewikkelde algoritmen, waarvan beheersing steeds weer problemen geeft, en die vaak worden aangeleerd in de vorm van regels met steeds verdergaande verbijzonderingen, te laten inoefenen door het expliciteren van het min of meer impliciet aangeleerde algoritme. In het aan de Subfaculteit Psychologie van de Vrije Universiteit lopende onderzoekprogramma met betrekking tot computer-gestuurd onderwijs wordt binnen een deelproject deze werkwijze toegepast. De leerlingen krijgen als op-

dracht om een denkbeeldige op het gebied van de werkvoorsvormen onwetende leerling de spellingsregels bij te brengen. Ze dienen daartoe via de computer-terminal geleidelijk aan het algoritme dat daarvoor nodig is te ontwikkelen.

Literatuur

Beishuizen, J. J. *Onderwijsstrategie en leerstrategie; een onderzoek naar de interactie tussen serialistische/holistische leerstrategie en volgorde van leerstofaanbieding in een algoritmisch leerprogramma syllogistisch redeneren*. Doctoraalscriptie Subfaculteit Psychologie, Vrije Universiteit, Amsterdam, 1975.

Boekaerts, M. *Towards a theory of learning based on individual differences*. Academisch proefschrift, Katholieke Hogeschool Tilburg, 1976.

Gal'perin, P. J. *De organisatie van de cognitieve activiteit en de optimalisering van het onderwijsleerproces*. (Vertaling uit het Russisch) *Pedagogische Studiën*, 1978, 55, 218-227.

Landa, L. N. *Algorithmierung im Unterricht*. Berlin: Volk und Wissen, 1969.

Landa, L. N. *Instructional regulation and control; cybernetics, algorithmization, and heuristics in education*. New Jersey: Englewood Cliffs, 1976.

Leeuw, L. de. *Computer gestuurd denken; training in het hanteren van algoritmische en heuristische probleemoplossingsmethoden*. *Pedagogische Studiën*, 1975, 52, 377-393.

Leeuw, L. de. *Leren probleemoplossen. Onderzoek naar het effect van het aanleren van algoritmische en heuristische oplossingsmethoden mede in verband met persoonskenmerken*. Lisse: Swets & Zeitlinger, 1979.

Parreren, C. F. van & J. A. M. Carpay. *Sovjetpsychologen*

*aan het woord*. Groningen: Wolters-Noordhoff, 1972.

Pask, G. & B. C. E. Scott. Learning and teaching strategies in a transformation skill. *British Journal of Mathematical and Statistical Psychology*, 1971, 24, 205-229.

Pask, G. & B. C. E. Scott. Learning strategies and individual competence. *International Journal of Man-Machine Studies*, 1972, 4, 217-253.

Resnick, L. B. Task analysis in instructional design: some cases from mathematics. In: D. Klahr (Ed.), *Cognition and instruction*. Hillsdale, New Jersey: Lawrence Erlbaum Assoc., 1976.

Selz, O. *Versuche zur Hebung des Intelligenzniveaus. Ein Beitrag zur Theorie der Intelligenz und ihrer erziehlischen Beeinflussung*. *Zeitschrift für Psychologie*, 1935, 134, 236-301.

Wolters, M. A. D. Algoritmen en aanschouwelijke schema's in het grammatika-onderwijs. *Pedagogische Studiën*, 1974, 51, 307-315.

Curriculum vitae

L. de Leeuw (1936) studeerde psychologie (hoofdrichting funktieleer) aan de Vrije Universiteit van 1961 tot 1967. Is vanaf 1969 werkzaam als wetenschappelijk medewerker bij de vakgroep Funktieleer (sectie cognitieve psychologie) van de Vrije Universiteit. Werkzaam op het gebied van het leren probleemoplossen. Promoveerde in 1979 op dit onderwerp.

Adres: Vrije Universiteit, Provisorium C 124, De Boelelaan 1115, Amsterdam.

Manuscript aanvaard 7-10-'80