

De computer en het onderwijsleerproces; een inleiding

L. DE LEEUW

*Vrije Universiteit, Vakgroep Functieleer en
Methodenleer, Amsterdam*

G. BEUKHOF

*Berenschot B.V., Adviesgroep Sociaal
Management, Utrecht*

Samenvatting

De computer in onderzoek van onderwijsleerprocessen mag zich verheugen in een groeiende belangstelling. In dit inleidende artikel op het themanummer wordt aandacht besteed aan de rol van de computer als instructiemedium en onderzoeksinstrument. Een uitwerking vindt plaats naar tutoriële systemen, de computer als onderzoeksinstrument en de 'intelligentie' van de tutoriële systemen. Het geheel wordt geïllustreerd met onderzoek, dat loopt in de Onderzoeksthema-groep (OTG) Onderwijsleerprocessen.

Introductie

In dit themanummer wordt verslag gedaan van onderzoek van onderwijsleerprocessen, waarbij de computer als instrument ingeschakeld wordt. Daarbij ligt de nadruk niet zozeer op het ontwikkelen van computerondersteunde onderwijsprogramma's zonder meer, gericht op praktische doeleinden, maar op theoretisch onderzoek, dat vergelijkend of toetsend van aard is. Het betreft onderzoek dat door de Stichting voor Onderzoek van het Onderwijs (mede-)gesubsidieerd is en dat via de OTG Onderwijsleerprocessen is beoordeeld en ingediend bij deze subsidiegever. Gezien de aard van de financieringsbron, betreft het wel steeds onderzoek dat ten minste potentiële relevantie heeft voor de onderwijspraktijk, soms op de langere termijn.

Formatieve evaluatie van subsidieaanvragen op dit gebied heeft plaats in de werkgroepen binnen de genoemde OTG. Een deel van

de lopende projecten is in de werkgroep Onderwijs en Informatietechnologie behandeld en geëvalueerd; andere projecten zijn via andere werkgroepen ingediend, bijvoorbeeld de werkgroep Cognitief Leren en Probleemoplossen, Verwerken van Teksten en Onderwijsleerproblemen.

Met nadruk zij gesteld, dat het hier gerapporteerde onderzoek niet alle lopende of afgesloten projecten op dit gebied betreft; het gaat om een selectie.

De groeiende belangstelling van onderzoekers voor de computer als instructiemedium en onderzoeksinstrument is gebaseerd op potentiële mogelijkheden van dat instrument voor individualisering in het onderwijs, remediëring van leerproblemen, bevordering van zelfstandige werkvormen en aanbidding van interactieve, te exploreren leeromgevingen, zogenaamde microwerelden waarin het leren door doen ('learning by doing') centraal staat.

In computerondersteund onderwijs is differentiatie naar tempo, niveau en leerstijl mogelijk, terwijl van volledig beheersingsleren sprake kan zijn. In dit inleidende artikel wordt kort ingegaan op de toenemende onderzoeksinteresse voor de computer als instructiemedium. Aandacht wordt besteed aan de mogelijkheden van geavanceerde tutoriële systemen, de gebruiksmogelijkheden van de computer voor het verrichten van systematisch (vergelijkend) onderzoek en de 'intelligentie' van tutoriële systemen. Zoveel mogelijk zal verwezen worden naar onderzoek en ontwikkelingswerk op het terrein van onderwijsleerprocessen (zie ook Beukhof & Simons, 1986).

Van lineaire, geprogrammeerde instructie naar intelligente tutoriële systemen

Onderwijs in het algemeen moet rekening houden met individuele verschillen tussen leerlingen, verschillen in begaafdheid (niveau), maar ook verschillen in geaardheid (leerstijlen, motivatie, concentratievermogen, faalangst). Op grond hiervan bestaat veel aandacht voor individualisering, ook binnen

het klassikale onderwijs. Geprogrammeerde instructie maakt, afhankelijk van de mate van geavanceerdheid, deze differentiatie mogelijk. Op de geavanceerdheidsdimensie dienen lineaire, geprogrammeerde instructies ter linkerzijde te worden gedacht en intelligente tutoriële systemen ter rechterzijde. Daar tussenin hebben dan vertakte geprogrammeerde instructies in klassieke vorm en gewoon (niet-intelligent) computerondersteund onderwijs hun plaats. Van links naar rechts op de dimensie worden steeds nieuwe mogelijkheden aan het onderwijs- of individualiseringsarsenaal toegevoegd. We zullen deze mogelijkheden in het kort aangeven.

In de lineaire, geprogrammeerde instructie wordt de leerstof aangeboden in kleine stappen. Steeds dient binnen een stap (schakel of frame) een antwoord te worden gegeven (het principe van het actief responderen). Deze stappen dienen zodanig klein te zijn, dat 90 à 95% van de leerlingen het gevraagde antwoord kan geven. Deze ideale stapgrootte komt pas tot stand door in een aantal try-out-en bijstellingscycli de moeilijkheidsgraad te optimaliseren. Een belangrijk kenmerk is dat de leerling directe feedback (niet altijd met uitleg) krijgt over het goed of fout zijn van het gegeven antwoord. Deze vorm van geprogrammeerde instructie, die overigens een grondige leerstofanalyse nodig maakt, kan in boekvorm, maar beter met een onderwijsmachine ('teaching machine') worden aangeboden. Alleen differentiatie naar tempo is mogelijk. Aanpassing van de stapgrootte bij begaafdheid (niveau) is niet mogelijk, mede een reden waarom deze instructievorm nauwelijks meer in gebruik is. Dit hangt samen met de 'dullheid' van de programma's: het bevat voor de niet-minimaal begaafde leerlingen (waarop de stapgrootte is afgestemd) te veel herhalingen en te kleine stappen.

In de vertakte, geprogrammeerde instructie is een redelijke mate van differentiatie naar niveau mogelijk (naast differentiatie naar tempo). De meest begaafde leerlingen hoeven niet in de vertakkingen terecht te komen en kunnen lineair door het uit wat grōtere stappen bestaande programma lopen. Het principe van het actief responderen blijft hier in wat gewijzigde vorm (actief kiezen maken) gehandhaafd. Het programma is meer remediërend van aard, omdat op gemaakte fouten kan worden ingegaan, uitleg kan worden verschaft

en misconcepties kunnen worden rechtgezet. In rudimentaire vorm is het programma diagnostisch van aard. De zekerheid dat leerlingen, zeker zij die veel vertakkingen hebben doorlopen, een desbetreffend stukje leerstof na afloop beheersen, is niet gegarandeerd. Van beheersingsleren ('mastery learning') is niet echt sprake (zelfs minder dan bij de lineaire vorm).

Computerondersteund onderwijs heeft het grote voordeel dat differentiatie naar tempo en niveau mogelijk is, terwijl van volledig beheersingsleren sprake kan zijn. Doordat de leergeschiedenis kan worden geregistreerd (tijd, gemaakte fouten van verschillend type, gevraagde hulp), kan over een reeks van opgaven worden bijgehouden in hoeverre aan een gesteld beheersingscriterium wordt voldaan. Bovendien kan met de leergeschiedenis (binnen een opgave en eventueel over een reeks van opgaven) rekening worden gehouden bij het geven van feedback (met uitleg) en bij het bieden van hints. Het diagnostisch vermogen van de desbetreffende programma's is groter dan bij de vormen die meer naar links op de dimensie staan. Dit vereist echter wel dat de programmaconstructeur voor elke opgave leerlingantwoorden, die op verschillende typen misvattingen of denkfouten berusten, voorziet. Wordt een van deze fouten gemaakt, dan is een specifieke feedback met uitleg mogelijk.

Ook kan de te bieden hulp multiconditioneel zijn, gebaseerd op meerdere voorafgaande antwoorden of hulpverzoeken binnen het desbetreffende probleem; anders gezegd: op kennis die het programma bezit over de door de leerling gemaakte fouten (een primitief leerlingmodel).

Omdat elke nieuwe opgave door de programmaconstructeur moet worden ingebracht in het programma en de beslissingen over de te nemen tutoriële acties voor elke opgave moeten worden geëxpliciteerd, kan van 'nietintelligent' computerondersteund onderwijs worden gesproken. Binnen intelligent computerondersteund onderwijs, ook intelligente tutoriële systemen of kennisgebaseerde systemen genoemd, wordt elke nieuwe opgave op grond van in het programma ingebouwde domeinkennis door het programma gegenereerd. De tutoriële (onderwijskundige) beslissingsregels zijn in de tutoriële component of expert

ingebouwd, vaak in de vorm van 'alsdan'-regels (vergelijk Newell & Simon, 1972). Vanaf het begin van het programma wordt een leerlingmodel opgebouwd. Deze kan de door de leerling over een reeks van opgaven gemaakte fouten bevatten: soort en frequentie, de tijd die nodig is voor (deel)processen, de benodigde hulp en het niveau waarop deze nodig is, de geconstateerde misconcepties. Met deze gegevens kan de tutor-expert dan weer rekening houden.

Het samenspel van onderwijskundige regels (het onderwijskundige produktiesysteem) dient goed van tevoren doordacht te worden (wat te doen als de leerling een weinig belovende oplossingsweg inslaat, wanneer in te grijpen, welk niveau van hulp (specifiek of algemeen) te bieden enz.), maar ligt daarna meestal vast. Er zijn zelfs meer geavanceerde vormen, waarbij van self-improving tutors sprake is (O'Shea, 1982; Kimball, 1982), dat wil zeggen van onderwijskundige experts, die zich zelf bijstellen. Bierman en Kamsteeg geven in dit themanummer een schema van de verschillende componenten van een intelligent tutorieel systeem. Buiten de reeds genoemde componenten (domeinexpert, leerlingmodel en tutoriële component) onderscheiden zij een diagnosecomponent en een interactiecomponent.

De computer als middel voor systematisch (vergelijkend) onderzoek

De toenemende populariteit (binnen de populatie van onderwijsonderzoekers) van de computer als medium heeft niet alleen betrekking op de directe onderwijskundige voordelen (individualisering door onder andere directe feedback, diagnostische remediëring, beheersingsleren), maar ook op de gebruiksmogelijkheden van de computer voor het verrichten van systematisch (vergelijkend) onderzoek. Het feit dat elke operatie van leerling en 'leerkracht' (tutor) en de interactie tussen beiden in het computerprogramma dient te zijn neergeslagen, impliceert dat het onderwijsleerproces op sterk geëxpliciteerde wijze verloopt en daardoor repliceerbaar is, terwijl door de mogelijkheid van registratie van het onderwijsleerproces naast produktiematen ook procesmaten beschikbaar komen.

Computerbestuurd vergelijkend onderzoek

vereist, in tegenstelling tot wat gebruikelijk is bij andersoortig vergelijkend onderzoek, detailmatige programmering. Zo dient goed nagedacht te worden over het begingedrag, dat aanwezig wordt verondersteld en hoe daarbij aan te sluiten, hoe de leerstof te structureren en volgens welke sequentie deze aan te bieden (wat de criteria dienen te zijn voor beheersing op de verschillende niveaus). Leerpsychologische theorieën, zoals die van Gagné (1986) of van Galperin (zie Van Parreren & Carpay, 1972) zijn hierbij relevant. Ook de vraag in welke mate de stof programmagestuurd of leerlinggestuurd ('program versus learner control') moet worden aangeboden, dient beantwoord te worden (Reigeluth, 1983).

Zoals gezegd, maakt registratie van procesgegevens een veel grondiger inzicht in het verloop van het leren mogelijk. Ook kan de analyse van relaties tussen leerlingkenmerken, leerprocesmaten en leerproduktmaten het theoretische inzicht verdiepen (zie bijvoorbeeld het onderzoek van De Leeuw, 1979, waarbij de effecten van het aanleren van algoritmische en heuristische oplossingsmethoden werden vergeleken).

Een tweede voorbeeld van vergelijkend onderzoek met behulp van computerbestuurd onderwijs is te vinden in het in dit themanummer beschreven onderzoek van Van Daal, Van der Leij, Bakker en Reitsma.

Naast vergelijkend onderzoek waarin didactische theorieën zijn betrokken, die op zich met het medium computer niets te maken hebben (bijvoorbeeld de effecten van regelleren versus ontdekkingsleren) en waarbij de computer uitsluitend als medium voor gecontroleerde presentatie wordt gebruikt, wordt vergelijkend onderzoek uitgevoerd dat op de specifieke mogelijkheden van het medium is gericht, zoals onderzoek naar de effecten van verschillende feedback-vormen, leerlingcontrole versus programmacontrole en aansluiting bij de stijl van informatieverwerken (bijvoorbeeld verbaal-visueel).

Dit komt tot uiting in de vijf didactische functies die binnen de werkgroep Onderwijs en Informatietechnologie van de OTG Onderwijsleerprocessen als beschrijvingskader worden gehanteerd, te weten:

1. Onderwijsvoorbereiding

In hoeverre is het mogelijk om op basis van interactieve kenmerken van computeron-

dersteund onderwijs het onderwijs te individualiseren door aan te sluiten bij de voorkeuren en de manieren van informatieverwerken van de leerlingen en wat zijn de effecten daarvan?

2. *Structurering, sequentiëring en presentatie*

Wat zijn de mogelijkheden en effecten van adaptieve ordening van de leerstof aan de individuele leerling? In welke mate dient er sprake te zijn van 'learner control' of van 'program control'? Welke presentatiemodi zijn mogelijk (grafisch, verbaal, auditief)?

3. *Realisatie van de taakruimte*

Hoe moet de taakruimte ingericht zijn, wil leren het gewenste resultaat opleveren? Bijvoorbeeld aan welke criteria moet de dialoogvoering tussen mens en machine in onderwijssituaties voldoen (vergelijk Norman & Draper, 1986; Shneiderman, 1987)?

4. *Feedback*

In welke mate is onmiddellijke, lokale feedback voor verschillende leerlingen gewenst ten opzichte van onverzichts-feedback of meer globale feedback?

5. *Evalueren en beslissen*

Hoe wordt het verloop van het onderwijsleerproces gestuurd? Wat is de mate van adaptiviteit? Welke beslissingsregels worden er gehanteerd voor de voortgang (bijvoorbeeld beheersingscriteria)? (Tennyson & Park, 1987)

Leerlinggestuurde instructie (ook wel 'learner control' genoemd) is een instructiestrategie waarbij belangrijke beslissingen (bijvoorbeeld met betrekking tot het studietempo, de volgorde van de te leren informatie en de beschikbaarheid van verschillende informatiebronnen) door de leerling zelf worden genomen. Bij programmagestuurde instructie bepaalt het programma hoe de instructie voor de leerling verloopt (Beukhof, 1987; Hasselerharm, 1987). Onderzoek naar leerlinggestuurde instructie (versus programmagestuurde instructie) leidt nog niet direct tot de eenduidige conclusie dat deze instructie een positief effect heeft op de leerresultaten. In sommige onderzoeken wordt zelfs aangetoond dat leerlinggestuurde instructie leidt tot slechtere prestaties (Hannafin, 1984). Onderzoek naar de effecten van combinaties van leerling- en programmagestuurde instructie, waarbij soms het programma de keuzen maakt - afhankelijk van bijvoorbeeld studievorderingen - en soms de leerling, aangepast aan de cognitieve

structuur, interesse en stijl, is nog weinig uitgevoerd.

De vraag in dat onderzoek luidt dan niet: wel of geen leerlinggestuurde instructie, maar: hoe, wanneer en op welke manier? Belangrijk hierbij is dat de leerling inzicht krijgt in de verschillende keuzemogelijkheden, die het programma hem/haar geeft. Training in het gebruik van leerlinggestuurde instructie is daartoe noodzakelijk. Voor ontwerpers van leerlinggestuurde instructie lijkt nog een lange weg te gaan (Duchastel, 1986).

Niet alleen voor het doen van vergelijkend onderwijskundig onderzoek is de computer een belangrijk hulpmiddel. Ook als cognitief psychologische theorieën of theorieën over samenwerkend leren op hun onderwijskundige relevantie worden getoetst, is de computer het medium bij uitstek om dit verantwoord te doen. De theorieën worden dan in de vorm van computerprogramma's gesimuleerd. Door vergelijking van het proces en het produkt van een echte leerling (of leerlingenaar) en het computerprogramma is dan toetsing van de theorie mogelijk. Voorbeelden hiervan bieden het in dit themanummer gerapporteerde onderzoek van Barnard, Erkens, Kanselaar en Van der Linden, en het hier niet-gerapporteerde SVO-onderzoek 'Training van aanpak en oplossing van rekenopgaven volgens een genetisch model' (zie Mannaerts, in druk).

De 'intelligentie' van tutoriële systemen

In een aantal van de bijdragen aan het themanummer is sprake van het geven van onderwijs c.q. het doen van vergelijkend onderwijspsychologisch onderzoek met behulp van intelligente tutoriële systemen, ook wel kennisgebaseerde systemen genoemd. Wat maakt een tutorieel systeem intelligent? Hoe verhoudt dit type onderwijssysteem zich tot niet (of minder) intelligente systemen? Op deze vraag zijn verschillende antwoorden mogelijk.

Vanuit de leerling beschouwd, geldt dat deze het systeem als zeer responsief zal ervaren (al zal hij/zij niet in deze termen hierover spreken of denken).

De leerling zal - als het systeem goed geconstrueerd is - de acties en reacties van het programma beschouwen 'alsof een individuele leerkracht hem/haar onderwijst en begeleidt'. Dit zal betekenen dat het intelligente

computerbestuurde programma op een fout niet reageert met feedback zoals 'jouw antwoord is fout', maar met bijvoorbeeld 'jouw antwoord is fout, je hebt dezelfde soort fout al twee keer eerder gemaakt, zou je niet eens om hulp vragen?' of 'jouw antwoord is goed, maar de manier waarop je dit antwoord gevonden hebt, is niet de meest efficiënte'.

Ook het feit dat niet alleen multiple choice of (korte) open-einde-antwoorden van de leerling door het programma kunnen worden geïnterpreteerd, maar dat ook tekeningen die de leerling op het scherm maakt door het programma (in combinatie met alfanumerieke antwoorden) kunnen worden geduid, vormt een aspect van een intelligent tutorieel systeem (in het vervolg ITS genoemd).

Een ander belangrijk aspect van een ITS is de automatische – random of systematische – generering van opgaven. Dit aspect zal weliswaar niet door de leerling ervaren worden, maar dat doet aan het intelligente karakter van het programma niets af. Uit een kennisdomein (bijvoorbeeld thermodynamica of grammatica) kunnen door het programma opgaven worden samengesteld – eventueel op grond van bepaalde nadere beslissingsregels die zijn afgeleid uit voorafgaande leerlingprestaties. Het programma kent bij elke opgave de juiste oplossingsmethode: de stappen die tot de oplossing moeten leiden en de eindoplossing die daaruit volgt. Als er meerdere aanpakken mogelijk zijn, dan kent het programma – dat als expert op het gebied geldt – deze. Dit impliceert dan weer dat leerlingstappen geïnterpreteerd kunnen worden – eventueel als behorend tot een van meerdere oplossingsmethoden.

Een ITS kan ook vastleggen welke typerende fouten gemaakt worden door een bepaalde leerling. De fouten kunnen gebaseerd zijn op misconcepties, misvattingen van bepaalde wetmatigheden of algoritmische regels.

Ook kan het zijn dat een leerling een eigen versie hanteert van een officieel algoritme, bijvoorbeeld van het algoritme voor optellen van getallen van drie cijfers, of staartdelingen of werkwoordsvormen. Dat deze systematische idiosyncratische algoritmen ('bugs') bestaan, is overtuigend aangetoond door Brown en Van Lehn (1980), en door Schoenfeld (1985). De bestaande 'bugs' kunnen vervolgens door het programma worden gerepareerd (geen si-

ecure overigens) of de leerling kan door de leerkracht, op grond van informatie uit het programma, op de desbetreffende punten worden 'herschold'. Dit aspect van ITS'en duidt op het diagnostische karakter ervan.

Het genoemde soort diagnose komt tot stand door over een reeks opgaven de aard van de gemaakte fouten te registreren en dan te vergelijken met aan het programma bekende typen fouten, die staan voor soorten afwijkingen in het officiële algoritme. Het programma kan deze foutentypen door de programmaconstructeur 'aangeleverd' hebben gekregen (deze heeft de fouten dan zelf 'berekend' door verschillende 'bugs' te veronderstellen). Intelligente programma's kunnen de foutentypen ook zelf 'berekenen' door systematisch allerlei 'verminderingen' aan te brengen in het algoritme en de consequenties hiervan door te rekenen (zie bijvoorbeeld Pijls & Kempen, in druk).

Een ITS is, zoals reeds vermeld, opgebouwd uit een aantal componenten. Allereerst is er een component die de domeinkennis (declaratieve en procedurele kennis van het leerstofgebied) bevat. Naast deze domeinexpertcomponent wordt de leerlingcomponent onderscheiden. Hierin wordt alle informatie opgeslagen over de leerling zoals die blijkt uit diens acties en reacties over een reeks van problemen: beheerste kennis, geconstateerde misvattingen etc. Ook kunnen daarbinnen gegevens over relevante leerlingkenmerken worden opgeslagen, zodat hiermee bij de presentatie van de leerstof rekening kan worden gehouden (bijvoorbeeld faalangst, behoefte aan structuring of een leerstijlgegeven als verbaliseerder/visualiseerder dat de preferentie van de leerling voor verbale versus figurele informatieverwerking aangeeft).

Een ander belangrijk bestanddeel van een ITS is de diagnostische component. Uit de discrepantie tussen de door de domeinexpert berekende oplossingsweg, ook wel normspoor genoemd, waarbij eventueel rekening wordt gehouden met de leerling-eigen wijze van aanpak van het probleem, en het in de leerlingcomponent geregistreerde oplossingsproces van de leerling, wordt afgeleid welk soort fout is gemaakt en welke mogelijke misconcepties of 'bugs' een rol kunnen spelen. Dit diagnostische proces zal dienen te leiden tot bijsturing van de leerling. Acties van de leerling, uit te voeren via het toetsenbord of in de vorm van

tekeningen (met behulp van de muis), moeten door het programma kunnen worden begrepen, zodat adequate reacties mogelijk zijn. Ook moeten responsies door het programma in voor leerlingen begrijpelijke vorm worden omgezet. Dit vertalen heeft plaats met behulp van de interactiecomponent van het programma.

De tutor-component (onderwijsexpert) bevat de didactische beslissingsregels, die een conditie-actievorm hebben. Zo'n regel schrijft bijvoorbeeld voor wat te doen als een leerling tweemaal een fout op hetzelfde punt heeft gemaakt en niet zelf om hulp heeft gevraagd; of: wat te doen zodra een leerling (op een bepaald punt) om hulp vraagt. Die regels kunnen multiconditioneel zijn. Het is goed de lezer erop te wijzen dat de introductie van ITS'en weliswaar een nieuwe ontwikkeling impliceert, maar dat er geen sprake is van een scherpe cesuur ten opzichte van vroegere, minder intelligente vormen van computerbestuurd onderwijs. Zo wordt zelfs in de drill-and-practice-vorm van computerondersteund onderwijs in rudimentaire zin een leerlingmodel opgebouwd. In deze onderwijsvorm wordt vaak geleerd tot een beheersingscriterium is bereikt. Het bijhouden van de stand van zaken op dit punt impliceert registratie binnen een 'leerlingmodel'. In tutoriële programma's (van de niet-intelligente soort) zijn de desbetreffende 'componenten' al minder rudimentair aanwezig. Leerlingen kunnen worden aangezet of gedwongen om hun oplossing op het probleem stapsgewijs te geven. Dit impliceert dat feedback, maar ook tutoriële ondersteuning, in elk stadium van probleemoplossen mogelijk wordt.

Wat in elk geval kenmerkend is voor een ITS is de generering van opgaven met de daarbij behorende oplossingsmethode door de domeinexpert en het op basis van de gang van zaken binnen een reeks opgaven opbouwen van een leerlingmodel en het op grond van beide vaststellen van misconcepties en 'bugs' aan de kant van de leerling.

Het moet overigens duidelijk zijn dat de plaats van de programmavormen op het continuum minder – meer intelligent geen verband houdt met de instructionele waarde van de programma's. Als we instructionele waarde van een onderwijsvorm definiëren als de mate waarin gestelde leerdoelen door deze onderwijsvorm bereikt worden, dan geldt dat een

drill-and-practice-programma een even grote instructionele waarde kan bezitten als een ITS. Een voorbeeld van een in potentie op het desbetreffende leerdoel (het leren lezen van woorden) toegesneden drill-and-practice-programma is het in de bijdrage van Van Daal c.s. beschreven, door hen ontwikkelde programma. Als de constructeurs erin slagen om het probleem van het geven van feed-back op de gevraagde verklanking op te lossen – niet eenvoudig overigens – dan is een programma met hoge instructionele waarde geboren.

Omgekeerd geldt, dat hoeft waarschijnlijk geen betoog, dat ITS'en niet per definitie een hoge instructionele waarde hebben. Ook als de programmaconstructeurs erin slagen om de domeinkennis correct te representeren, ook als de meest geavanceerde interacties tussen leerling en programma mogelijk zijn, dan nog is bereiking van de leerdoelen niet gegarandeerd. De tutoriële component vormt de achilleshiel van dit soort programma's. Niet voor niets wordt in het artikel van Bierman en Kamsteeg zoveel aandacht aan de bouw van deze component geschonken. Ook de in de bijdrage van De Leeuw, Beishuizen, Van Daalen, Meyer en Perrenet beschreven problemen bij het bieden van hulp en coaching hebben op tutoriële aspecten betrekking.

Tot besluit

Voor de ontwikkeling van computerondersteund onderwijs en ITS'en is verdere theorievorming noodzakelijk (zie het programma voor onderzoek van de OTG Onderwijsleerprocessen, Beukhof & Simons, 1986), onder andere naar:

- leerprocessen met betrekking tot automatisering, begripsvorming, probleemoplossen en het verloop van dergelijke processen bij leerlingen van verschillende leeftijd bij verschillende leerstofgebieden;
- instructieprocessen, zoals deze zijn uitgewerkt naar de reeds eerder genoemde didactische functies;
- motivatieprocessen in relatie tot computergebruik in het onderwijs (zie met name Keller & Suzuki, 1987);
- veranderingen in het docentgedrag door invoering van de computer.

Het moge duidelijk zijn dat de ontwikkelingen nog in volle gang zijn en dat daarbij

personen uit verschillende disciplines nationaal en internationaal samenwerken (Moonen & Plomp, 1987).

Literatuur

- Beukhof, G., *Tekstbestudering: tekst- en studentgecontroleerd*. Bijdrage aan het Psychologencongres, Groningen, 6 en 7 november 1987.
- Beukhof, G. & P. R. J. Simons (Red.), *Onderwijsleerprocessen: een programma voor onderzoek*. Enschede: OTG Onderwijsleerprocessen/ Universiteit Twente, oktober 1986.
- Brown, J. S. & K. van Lehn, Repair theory: a generative theory of bugs in procedural skills. *Cognitive Science*, 1980, 4, 379-426.
- Duchastel, P., Intelligent computer assisted instruction systems: the nature of learner control. *Journal of Educational Computing Research*, 1986, 2, 379-393.
- Gagné, R. M. (Ed.), *Instructional technology: foundations*. Hillsdale, N.J.: Lawrence Erlbaum, 1986.
- Hannafin, M. J., Guidelines for using locus of instructional control in the design of computer assisted instruction. *Journal of Instructional Development*, 1984, 7, 6-10.
- Hasselerharm, E., *Leerlingsturing in een computergestuurde instructie*. Enschede: Universiteit Twente, 1987.
- Keller, J. M. & K. Suzuki, Use of the ARCS-motivation model in courseware design. In: D. H. Jonassen (Ed.), *Instructional designs for micro-computer courseware*. Hillsdale, N.J.: Lawrence Erlbaum, 1987.
- Kimball, R., A self improving tutor for symbolic integration. In: D. Sleeman & J. S. Brown (Eds.), *Intelligent tutoring systems*. London: Academic Press, 1982.
- Leeuw, L. de, *Leren probleemoplossen*. Lisse: Swets & Zeitlinger, 1979.
- Mannaerts, A. A. J., Een instrument voor de diagnose van rekengedrag. *Nederlands Tijdschrift voor de Psychologie* (in druk).
- Moonen, J. & T. Plomp (Eds.), *EURIT 1986: Developments in educational software and courseware*. Oxford: Pergamon Press, 1987.
- Newell, A. & H. A. Simon, *Human problem solving*. Englewood Cliffs: Prentice Hall Inc., 1972.
- Norman, D. A. & S. W. Draper (Eds.), *User centered system design: new perspectives on human computer interaction*. Hillsdale, N.J.: Lawrence Erlbaum, 1986.
- O'Shea, T., A self improving quadratic tutor. In: D. Sleeman & J. S. Brown (Eds.), *Intelligent tutoring systems*. London: Academic Press, 1982.

- Parreren, C. F. van & J. A. M. Carpay, *Sovjet-psychologen aan het woord*. Groningen: Wolters-Noordhoff, 1972.
- Pijls, F. & G. Kempen, Kennistechnologische leermiddelen in het grammatica- en spellingsonderwijs. *Nederlands Tijdschrift voor de Psychologie* (in druk).
- Reigeluth, C. M. (Ed.), *Instructional-design theories and models: an overview of their current status*. Hillsdale, N.J.: Lawrence Erlbaum, 1983.
- Schoenfeld, A. H., *Mathematical problem solving*. Orlando, Florida: Academic Press, 1985.
- Shneiderman, B., *Designing the user interface: strategies for effective human-computer interaction*. Reading: Addison-Wesley, 1987.
- Tennyson, R. D. & O. C. Park, Artificial intelligence and computerbased learning. In: R. M. Gagné (Ed.), *Instructional technology: foundations*. Hillsdale, N.J.: Lawrence Erlbaum, 1987.

Curricula vitae

L. de Leeuw studeerde psychologie aan de V.U. Was van 1967 tot 1969 als onderzoeker werkzaam bij het Nutsseminarium voor Pedagogiek. Vanaf 1969 werkzaam bij de Vakgroep Functieleer en Methodenleer van de V.U.; vanaf 1986 als universitair hoofddocent. Doet onderzoek op het gebied van het leren probleemoplossen en de meting van leerlingkenmerken. Promoveerde in 1979 op het leren probleemoplossen. Is supervisor van een aantal S.V.O.- en eerste geldstroomprojecten.

G. Beukhof (1949) studeerde Onderwijskunde, specialisatie 'Onderwijzen en Leren', aan de Universiteit van Utrecht. Van 1980 tot 1987 was hij werkzaam bij de Vakgroep Instructietechnologie van de Faculteit Toegepaste Onderwijskunde aan de Universiteit Twente. Vanaf 1979 was hij coördinator van de landelijke (SVO) onderzoeksthema-groep Onderwijsleerprocessen. Hij bereidt een proefschrift voor over het structureren en presenteren van instructieve teksten. Vanaf 1 juli 1987 is hij werkzaam bij Berenschot B.V., Sociaal Management, als senior-opleidings- en organisatie-adviseur.

Correspondentieadres: L. de Leeuw (zie p. 363). G. Beukhof, Berenschot B.V. Sociaal Management, Bernadottelaan 13, 3527 GV Utrecht. Tel. 030-916916.

Manuscript aanvaard 10-7-'87

Summary

L. de Leeuw & G. Beukhof. 'The computer in research on learning and instruction; an introduction.' *Pedagogische Studiën*, 1987, 64, 327-334.

There is a growing interest for the computer in research on learning and instruction. In this introduction, attention is paid to the computer as an instructional medium and as a research tool. A description is given of tutorial systems, the computer as a research tool, and the 'intelligence' of tutorial systems, with several illustrations from researchprojects of the Dutch Special Interest Group 'Learning and Instruction'.