

# Computergestuurd spellingonderwijs voor de basisschool

## 1. Inleiding

Voor diverse leerstofonderdelen van het moedertaalonderwijs worden de laatste jaren Computer Ondersteunde Onderwijs (COO) programma's ontwikkeld. Enkele voorbeelden zijn het COSO-project waarin een schrijfmgeving ontwikkeld wordt voor het stelonderwijs (Van der Geest, 1987) en de ontwikkeling van een intelligente omgeving voor computerondersteund grammatica- en spellingonderwijs in Nijmegen (Pijls & Kempen, 1987) en in Brussel (Daelemans, 1987).

Met betrekking tot het spelling- en grammatica-onderwijs zijn ook al verschillende COO-programma's op de markt verschenen. Veel daarvan zijn drill and practice programma's. In deze programma's ligt de nadruk op het extra oefenen van vaardigheden die de leerling nog niet voldoende beheerst. De leerling-antwoorden worden veelal voorzien van minimale feedback (goed/fout). Het bijbrengen van nieuwe kennis en vaardigheden komt niet of nauwelijks aan de orde.

Op dit moment ontwikkelen wij in het kader van het SVO-project "Feedbackprocessen bij computergestuurd spellingonderwijs" een tutorieel COO-programma voor het leren van de Nederlandse werkwoordsspelling. Hierin ligt het accent op het aanleren van nieuwe kennis en vaardigheden door middel van expliciete instructie in combinatie met gerichte oefening. De leerling-antwoorden worden veelal voorzien van informatieve feedback (waarom is een antwoord goed of fout). Het doel is het gecomputeriseerde onderwijsleesproces zo in te richten, dat de leerling zich de werkwoordsspelling zelfstandig eigen kan maken. Het programma wordt ontwikkeld voor MS-Dos machines en geprogrammeerd in Turbo Pascal.

In dit artikel gaan we in op leerstofsequentering, de instructiestrategie en verschillende onderwijspsychologische aspecten die in belangrijke mate de opzet van het instructieprogramma bepaald hebben. Daarna volgt een kort verslag van onze eerste bevindingen met het programma in proefscholen.

## 2. Inhoudelijke schets van het instructieprogramma

Elk COO-programma moet volgens ons gebaseerd zijn op een grondige leertaakanalyse. Uit een dergelijke analyse moet blijken over welke kennis en vaardigheden een leerling moet beschikken om de leertaak met succes te kunnen volbrengen. Deze analyse resulteert uiteindelijk in een leerstofordening waarin aangegeven wordt welke leerstofonderdelen op welk moment in het programma behandeld worden en welke begrips- en regelkennis bij elk leerstofonderdeel noodzakelijk is.

Het instructieprogramma is gebaseerd op recent onderzoek op het gebied van de werkwoordsspelling. Uit onderzoek (Assink, 1983; Zuidema, 1988) is gebleken dat het leren van de werkwoordsspelling via een algoritmische regelaanpak tot betere resultaten leidt dan de traditionele analogie-aanpak. In de analogiemethode wordt een beperkt aantal grondwerkwoorden behandeld aan de hand waarvan werkwoordsvormen analoog gespeld kunnen worden. In de algoritmische methode wordt het bijbrengen van de fundamentele begripsmatige kennis die nodig is om de systematiek in de spellingregels te doorzien, gekoppeld aan het leren van de schrijfregele. Centraal staat een algoritme voor het schrijven van de werkwoordsvormen. Deze strategie is hoofdzakelijk gebaseerd op Landa's theorie van het aligmiseren van leerprocessen (1974). Een algoritme is een handelingsvoorschrift dat bij correct gebruik een goede oplossing verzekert. Het functioneert voor de leerling als een hulpmiddel om het probleem van de werkwoordsspelling op te lossen, maar vooral ook als middel om het denken van de leerling te structureren en te sturen. Het algoritme tracht een klein deel van het taalsysteem voor de leerling inzichtelijk te maken.

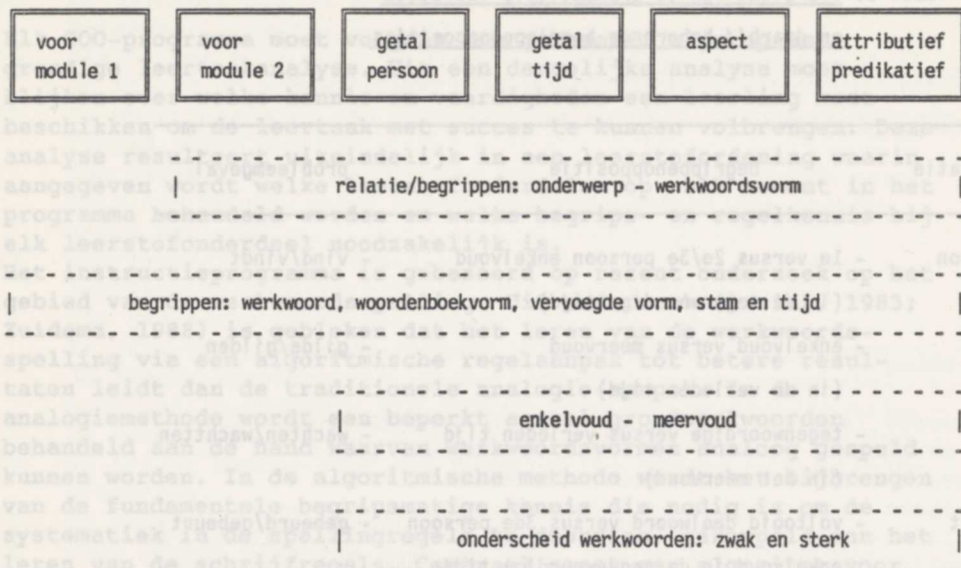
Op basis van het bovengenoemde onderzoek is de spellingmethode "De Werkwoordwinkel" ontwikkeld (Assink & Klein, 1984). Het algoritme voor het spellen van de werkwoordsvormen dat in deze methode centraal staat, bevat een aantal tweekeuze beslissingsknopen. Voorbeelden hiervan zijn: het onderscheid tussen de 1e en 2e/3e persoon enkelvoud in de tegenwoordige tijd, dat bepalend is voor het orthografische onderscheid vind/vindt en het onderscheid tussen de tegenwoordige en verleden tijd in het meervoud, dat bepalend is voor het orthografische onderscheid wacht/wachtten. Wanneer de leerling bij elk keuzemoment de juiste beslissing neemt, komt hij uiteindelijk bij de juiste schrijfwregel terecht. Deze begrippenopposities hebben betrekking op zes grammaticale operaties die de basis vormen voor de leerstofstructurering in het COO-programma (zie Tabel 1).

Tabel 1. Overzicht van de grammaticale operaties  
en daarbij behorende begrippenopposities

operatie	begrippenoppositie		probleemgeval
1. persoon	- 1e versus 2e/3e persoon enkelvoud (in de tegenwoordige tijd)	- vind/vindt	
2. getal	- enkelvoud versus meervoud (in de verleden tijd)	- gilde/gilden	
3. tijd	- tegenwoordige versus verleden tijd (in het meervoud)	- wachten/wachtten	
4. aspect	- voltooid deelwoord versus 3de persoon enkelvoud in de tegenwoordige tijd	- gebeurd/gebeurt	
5. vorm	- idem 4 in passieve zinnen		
6. attributief/ predikatief	- bijvoeglijk gebruikt voltooid deelwoord versus 3de persoon enkelvoud in de verleden tijd	- beantwoorde/ beantwoordde	

Het baseren van het instructieprogramma op deze grammaticale onderscheidingen vereist een rangordening van de operaties op grond van moeilijkheidsgraad. Uit empirisch onderzoek (Assink, 1987; Kattenberg, 1987; Verhoeven, 1985) is gebleken dat de operaties persoon, getal en tijd in deze volgorde het gemakkelijkst zijn. Aspect, vorm en attributief/predikatief zijn gemiddeld genomen het moeilijkst. Op basis van dit onderzoek is de leerstofinhoud van de lesmodulen van het instructieprogramma in hoofdlijnen vastgesteld (zie Figuur 1.).





Figuur 1. Globale leerstofinhoud van de zes lesmodulen

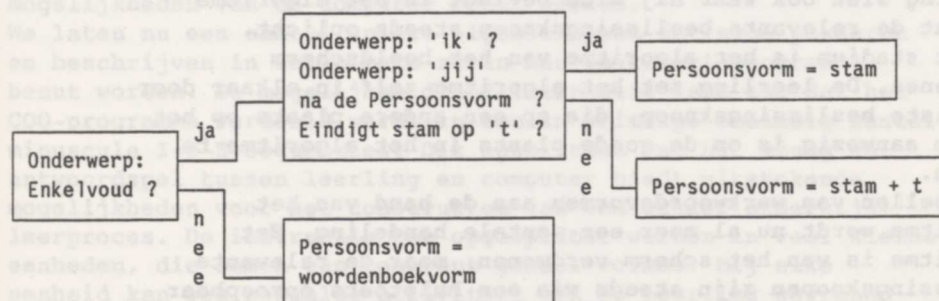
In Figuur 1 wordt de kern gevormd door de zes lesmodulen bovenaan. Deze volgen elkaar in de tijd op, te beginnen bij voormodule 1. Elk module vormt op zich een afgerond geheel. Als de leerling nog geen systematische instructie heeft gehad in de werkwoordsspelling, zal hij hooguit enige intuïtieve kennis hebben. Het programma moet aansluiten bij datgene wat de leerling al weet. Deze intuïtieve kennis vormt echter geen solide basis om direct de leertaak te kunnen uitvoeren. Daarom wordt in de eerste twee lesmodulen de noodzakelijke voorkennis aangebracht.

De eerste module, die nog vooraf gegaan wordt door een introductiemodule, geeft een globale oriëntatie op het begrip werkwoord. In de tweede module komt de relatie tussen het onderwerp en de persoonsvorm aan bod. Daarna wordt in elke module één grammaticale operatie of een combinatie van begrippenopposities aan de orde gesteld.

In de balken zijn de relatie, begrippen en onderscheidingen weergegeven die betrekking hebben op de lesmodulen die er ruimtelijk gezien recht boven staan. De begrippen woordenboekvorm (= infinitief) en/of stam komen bijvoorbeeld in

alle zes lesmodulen op een bepaalde manier aan bod. Op deze manier kan men uit Figuur 1 opmaken welke relaties, begrippen en onderscheidingen in elke lesmodule aan bod komen. De mate waarin deze begrippen behandeld worden, hangt af van het leerstofonderdeel in de desbetreffende module en kan als gevolg daarvan verschillen. Deze verschillen zijn niet zichtbaar in Figuur 1.

De leerling leert het spellen van de werkwoordsvormen in het programma aan de hand van de eerdergenoemde spellingalgoritme. Dit algoritme wordt op het scherm gepresenteerd, maar niet in z'n geheel zoals in de boekvorm, maar in delen. Elke grammaticale operatie komt in een bepaalde tak van het algoritme naar voren. Dat deel wordt in eerste instantie steeds geïsoleerd aangeboden. Het leren hanteren van het volledige algoritme verloopt daardoor in fasen. Als voorbeeld presenteren we het deelalgoritme waarmee in lesmodule 3 gewerkt wordt. Voor een weergave van het volledige algoritme verwijzen wij naar de bijlage.



Figuur 2. Het deelalgoritme uit lesmodule 3

Vanaf lesmodule 3 oefent de leerling eerst de betreffende grammaticale operatie met het relevante deel van het algoritme. Vervolgens wordt vanaf lesmodule 4 elk nieuw deel van het algoritme gekoppeld aan het behandelde deel en wordt het toegepast op reeds geoefende probleemttypen, die nu gemengd worden aangeboden. Via deze gefaseerde aanbieding wordt het eerste deelalgoritme uit lesmodule 3 uitgebreid tot het volledige algoritme in lesmodule 6.

Om het algoritme uiteindelijk zonder visuele en verbale ondersteuning op het scherm toe te kunnen passen wordt in elke lesmodule getracht het algoritme stapsgewijs te laten interioriseren. Aanvankelijk is het algoritme compleet met uitleg op het scherm aanwezig. Via programmasturing neemt de



leerling de juiste stappen in het algoritme en beslist wat het antwoord is bij elke beslissingsknoop. De programmasturing gaat geleidelijk over in leerlingsturing door gerichte oefening en verminderde verbale en visuele ondersteuning. In elke lesmodule onderscheiden we hierbij de volgende stadia:

1. Stapsgewijze uitleg en visuele ondersteuning bij elke stap in het algoritme.
2. Het toepassen van het algoritme op het desbetreffende probleemtype via programmasturing. De computer stelt de leerling de relevante vragen. De leerling antwoordt en krijgt informatieve feedback. De leerling ziet ook waar hij zich bevindt in het algoritme doordat de relevante beslissingsknoop steeds oplicht.
3. Bij het spellen van de werkwoordsvormen is het algoritme nog wel zichtbaar, maar de leerling krijgt in dit stadium geen uitleg meer. Hij moet zichzelf de relevante vragen stellen aan de hand van de beslissingsknoop die oplicht. Alleen bij een fout antwoord krijgt hij informatieve feedback. De leerling ziet ook waar hij zich bevindt in het algoritme doordat de relevante beslissingsknoop steeds oplicht.
4. In dit stadium is het algoritme van het beeldscherm verdwenen. De leerling zet het algoritme zelf in elkaar door de juiste beslissingsknoop, die op een andere plaats op het scherm aanwezig is op de goede plaats in het algoritme te zetten.
5. Het spellen van werkwoordsvormen aan de hand van het algoritme wordt nu al meer een mentale handeling. Het algoritme is van het scherm verdwenen, maar de relevante beslissingsknopen zijn steeds via een hulptoets oproepbaar voor de leerling. Bij een goed antwoord verschijnt het desbetreffende deel van het algoritme op het scherm en moet de leerling zelf verder. Bij een fout antwoord verschijnt informatieve feedback.
6. In het laatste stadium wordt bijna volledig op mentaal niveau met het algoritme gewerkt. Het algoritme is niet zichtbaar. De leerling kan bij elk oefenitem kiezen uit <1> direct het eindantwoord geven of <2> stap voor stap werken. Kiest hij voor <2> dan gaat hij even terug naar stadium 5.

Na deze zes stadia komt de leerling in een testfase waarin hij alleen op mentaal niveau gebruik kan maken van het algoritme. Hoe verder de leerling komt in het COO-programma, hoe meer de beslissingsknopen in het algoritme in verkorte vorm weergegeven worden. Voorbeelden hiervan zijn de begrippen onderwerp en persoonsvorm die verkort worden tot 'O' en 'P'. Op het moment dat deze lettersymbolen worden geïntroduceerd in het algoritme heeft de leerling er al ruimschoots mee gewerkt. Deze verkorting is enerzijds aangebracht uit ruimtelijke overwegingen, anderzijds wordt getracht een zekere verkorting in de mentale

representatie bij de leerling te bewerkstelligen. Op deze manieren proberen we het totale algoritme bij de leerlingen te interioriseren, waardoor ze over een middel beschikken om werkwoordsvormen correct te kunnen spellen.

### 3. Onderwijspsychologische aspecten bij het ontwerpen van COO

Kenmerkend voor het gangbare onderwijsleerproces is de I-O-E-sequentie. Eerst vindt een stuk instructie (I) plaats door de leerkracht, gevolgd door een oefenperiode (O). Daarna worden, vaak enige tijd na de oefenperiode, de leerresultaten door de leerkracht geëvalueerd (E). Tenslotte wordt, vaak enige tijd na het evaluatiemoment, de leerling geïnformeerd over het leerresultaat. Wanneer deze sequentie omgezet wordt in een gecomputeriseerd onderwijsleerproces, verandert er niets aan de formele structuur van deze sequentie maar wel aan de leerweg die de leerling aflegt. Op dat moment wordt het mogelijk om het onderwijsleerproces te optimaliseren, tenminste als de unieke mogelijkheden van de computer benut worden.

We laten nu een aantal unieke eigenschappen de revue passeren en beschrijven in welke mate ze in het instructieprogramma benut worden. De mogelijkheid tot interactie laat toe dat het COO-programma verdeeld wordt in een in principe oneindig aantal minuscule I-O-E sequenties. Dit opsplitsen van het vraag en antwoordspel tussen leerling en computer biedt uitstekende mogelijkheden voor het construeren van een actief onderwijsleerproces. De instructie kan opgesplitst worden in veel kleine eenheden, die samen een coherent geheel vormen. Bij elke eenheid kan een bepaalde activiteit van de leerling gevraagd worden. Een *tweede unieke eigenschap* is de beschikbaarheid van gedetailleerde antwoordgegevens. Deze informatie kan gebruikt worden om elk antwoord te bezien in het kader van het voortgaande leerproces. Bij elk antwoord kan feedback gegeven worden. Het verschaffen van *directe feedback* is de *derde unieke eigenschap*. De voordelen hiervan zijn dat de leerling direct na het uitvoeren van een beslissing geïnformeerd wordt over de juistheid ervan, maar vooral ook dat de feedback zo ingericht kan worden dat de leerling snel kan overgaan tot correctieve handelingen. Dit bevordert een efficiënt verloop van het leerproces. Tevens kunnen de oefenitems afgestemd worden op het behaalde leerresultaat. De leerling oefent dan geen overbodige items. Hierbij dringt zich wel de vraag op welk criterium men moet nemen om te beoordelen of een item overbodig is of niet. Naast de mogelijkheid tot efficiënte aanbieding van de items, is de totale hoeveelheid oefening die door de computer gegenereerd kan worden welhaast onuitputtelijk. Deze *vierde eigenschap* kan ook als uniek voor de computer beschouwd worden. Een gecomputeriseerd onderwijsleerproces, waarbij het programma



de rol van tutor heeft, vereist een explicitering van elke stap van de leerweg. Het programma vormt eigenlijk een afspiegeling van het onderwijsleerproces dat zich afspeelt tussen leerlingen en leerkracht. Elke leerstofstap die een leerkracht neemt in een onderwijssituatie moet expliciet in het programma aanwezig zijn. Hier ligt een essentieel verschil tussen pure oefenprogramma's en tutoriële COO. Een voordeel is dat het verloop van het leerproces daardoor dichter benaderd en systematischer gecontroleerd kan worden. Hoewel dit op zich gunstig is, levert een dergelijke explicitering ook problemen op. De ontwikkelaar moet zich bij elke instructie-oefening- evaluatie sequentie afvragen welke kennis en vaardigheden de leerling tot op dat moment in het programma verworven zou moeten hebben. De daarop volgende sequenties moeten dan zo ingericht worden dat de leerling zelfstandig de volgende fase in het leerproces kan bereiken. De kans van slagen van een dergelijke opzet hangt voor een deel af van de mate waarin de leertaak en de instructie goed te structureren of te algoritmiseren zijn. Het domein van de werkwoordsspelting beantwoordt aan dit criterium. Voor andere leerstofdomeinen, zoals begrijpend lezen, zijn gecomputeriseerde instructieprogramma's waarschijnlijk veel moeilijker te ontwikkelen.

Het bewerkstelligen van een efficiënt verloop van het leerproces is niet alleen op een oefening of een lesmodule toepasbaar maar ook op het gehele programma. Voorafgaand aan het werken met het programma wordt de voorkennis vastgesteld. Aan de hand daarvan bepaalt het programma welke lesmodule(n) door de leerling doorlopen moet(en) worden. Ook persoonlijkheidskenmerken (bijvoorbeeld faalangst) kunnen aanleiding zijn om de leerling alleen de op zijn situatie toegespitste leerstofonderdelen te laten doornemen. Kwalitatief verschillende instructiestrategieën kunnen geïmplementeerd worden. Hierdoor neemt de complexiteit van het programma sterk toe. Een vijfde unieke eigenschap is dan het vermogen van het programma om een meer op het individu afgestemde leerweg aan te bieden. Deze eigenschap staat zeker niet los van de andere maar berust eerder op een combinatie ervan. Een dergelijke geïndividualiseerde instructie zal in zijn meest ideale vorm voor elke leerling de optimale leerweg moeten kunnen uitstippelen. Daarbij zal het programma moeten beschikken over verschillende informatiebronnen. Het leerproces moet zowel vóór als tijdens de taakuitvoering gemeten worden (Tennyson, Christensen & Park, 1984). Een grote hoeveelheid meetmomenten zorgt voor de leergeschiedenis of het studentmodel. Op basis daarvan kan steeds de instructie zowel kwantitatief als kwalitatief aangepast worden. Hierbij wordt dan gebruik gemaakt van een expertsysteem en verschillende instructiestrategieën. Via een tweezijdige interactie kan de leerling ook vragen stellen aan



het programma, waardoor leerstof gegenereerd kan worden waaraan de leerling behoefte heeft. Een programma met dergelijke mogelijkheden wordt wel Intelligent Computer Assisted Instruction, ICAI genoemd. Deze benadering, waarbij technieken uit de Artificiële Intelligentie gebruikt worden, lijkt enorme mogelijkheden te bieden maar zal, zeker als toepassing in het reguliere basis- en voortgezet onderwijs, voorlopig nog wel een utopie blijven. Naast veel meer kennis over de inrichting van gecomputeriseerde onderwijsleerprocessen vereist zo'n programma een bijzondere ontwikkelsituatie met uitgebreide faciliteiten. Het is volgens Dirkzwager & Mol (1987) echter zeer wel mogelijk om ook zonder gebruikmaking van AI-technieken goed, "intelligent" COO te maken. Ook Bork (1987) is deze mening toegedaan.

In het door ons ontwikkelde tutoriële programma worden de genoemde unieke eigenschappen van de computer zoveel mogelijk benut. We streven echter geenszins naar ICAI. In eerste instantie is het onze bedoeling een programma te maken waarin de leerstof volgens een expliciete instructiestrategie (het algoritme) gepresenteerd wordt. Via kleine I-O-E-sequenties wordt de leerling stap voor stap door de leerstof geleid. Bij elk instructiemoment is gerichte oefening volgens ons noodzakelijk. Zodra het mogelijk is om de leerling via een oefening bij de stof te betrekken wordt dat gerealiseerd. Enkele voorbeelden van oefenvormen zijn: het classificeren, identificeren, bedenken en vervoegen van werkwoordsvormen, het aangeven van verschillen tussen relevante zinnen, zinsdelen of woorden. Vanwege het essentiële belang van grammaticale kennis staat in bepaalde delen van het programma het manipuleren met woorden en zinsdelen centraal. Het toepassen van grammaticale begrippen uit het algoritme vindt veelvuldig plaats. Ook wordt van de leerling in principe altijd een actieve inbreng verwacht. Het gebruik van multiple choice vragen komt nauwelijks voor. De antwoordmogelijkheden zijn veelal niet zichtbaar op het scherm. Hierdoor kan de leerling het antwoord niet herkennen maar moet hij het zelf genereren. Voorbeelden zijn het intypen van semantisch passende en orthografisch correcte werkwoordsvormen, het intypen van zelfbedachte infinitieven ("de woordenboekvorm"), het genereren van "de woordenboekvorm" bij vervoegde vormen en, omgekeerd, het genereren van de bijbehorende vervoegde vormen bij een vervoegde vorm. Dit betekent wel dat het programma bij een aantal oefeningen adequaat moet kunnen reageren op een diversiteit aan leerling-antwoorden. Het veelvuldig zelf intypen van werkwoordsvormen door de leerling brengt met zich mee dat er allerlei verschillende typ- en spelfouten gemaakt worden. Regelmatig voorkomende fouten zijn inversies, verschillende verwisselingen zoals au/ou, f/v, z/s en ei/ij, fouten tegen de consonant-vocaalregel, fouten die verband houden met de werkwoords-

vervoegingen (vooral wanneer leerlingen de systematiek in de werkwoordsspelling nog niet goed doorzien). Aangezien deze fouten worden gemaakt tijdens het leren spellen van werkwoordsvormen, zijn ze bijzonder interessant en moeten ze ook gezien worden in het kader van dat leerproces. Wanneer een leerling nu één of een combinatie van deze fouten maakt, maar wel het goede antwoord bedoelt (het programma bepaalt zelf deze grens), dan speelt het programma daar in de feedback op "intelligente" wijze op in (zie Figuur 3).

---

Jouw antwoord, gehaamert is niet helemaal goed.

Bij de gele letters is nog iets fout.

---

### Figuur 3. Feedbacksituatie

De onderstreepte letters in Figuur 3 worden in een afwijkende kleur weergegeven. Deze feedback geeft aan waar de fout zich in het antwoord bevindt, zonder daarbij te zeggen wat het goede alternatief is. Een dergelijke feedback is zinvol, ten eerste omdat een in principe fout antwoord positief benaderd wordt en ten tweede omdat het de leerling stimuleert zelf de aard van de fout te ontdekken. De leerling reflecteert zodoende op de gemaakte fout(en). Ter bevordering van het leerproces lijkt ons deze aanpak erg nuttig. Bovendien blijkt ze in de praktijk goed te werken. Kennis over de eerder genoemde typ- en spelfouten is opgenomen in een procedure die ervoor moet zorgen dat op een adequate manier op fouten in leerlingantwoorden wordt gereageerd. Deze procedure moet in ieder geval toegepast kunnen worden op alle werkwoordsvormen die het programma kan verwerken (ongeveer 14.000).

Via een rijke variatie in oefen- en spelvormen willen we een onderwijsleerproces realiseren waarbij de leerling niet de kans krijgt een passieve houding aan te nemen. Het programma maakt hierbij gebruik van een aantal databestanden. Hierdoor wordt het mogelijk oefeningen steeds weer met andere items, eventueel met bepaalde kenmerken, te laten uitvoeren.

Wanneer het programma losse werkwoordsvormen als oefenstof aanbiedt, kiest het daarbij uit een databestand waarin in gecodeerde vorm ongeveer 2000 infinitieven staan. Dit databestand vormt een benadering van de werkwoordenschat van onze doelgroep (eindjaren basisschool/begin voortgezet onderwijs). Dit bestand wordt door het programma omgezet in een "werkwoordenboek" dat op het scherm gepresenteerd wordt in een



zogenaamd "window". Op verschillende momenten kan dit werkwoordenboek door de leerling geraadpleegd worden. Voor de omzetting van dit bestand hebben we een flexibele procedure ontwikkeld waarin de benodigde kennis over het vervoegen van werkwoorden opgenomen is. Binnen deze omvangrijke procedure wordt een aantal subprocedures onderscheiden die werkwoorden elk op een andere manier vervoegen. De kennis die in deze subprocedures opgenomen is bestaat onder andere uit orthografische kenmerken die essentieel zijn voor diverse categorieën werkwoordsvervoegingen in het Nederlands (te vergelijken met de grondwerkwoorden uit de analogiemethode). Onder andere is kennis aanwezig over de consonant-vocaal regel, sterk of zwak werkwoord, z/s en v/f verwisselingen, prefix of geen prefixwerkwoord (verbranden versus branden). Ook zijn er specifieke orthografische kenmerken opgenomen die moeilijk te categoriseren vervoegingen opleveren. De flexibele procedure kan op grond van al deze taalkennis bij elke gecodeerde woordenboekvorm uit het databestand zelf de bijbehorende vervoegde werkwoordsvormen produceren. Dit betekent dat op elk willekeurig moment in het programma een bepaalde werkwoordsvorm, met of zonder een bepaald kenmerk, door het programma aangeboden kan worden. Het is zelfs zo dat een flink aantal niet in het databestand opgenomen infinitieven ook feilloos door de procedure vervoegd kunnen worden.

Naast dit bestand met gecodeerde infinitieven gebruikt het programma bestanden waarin zinnen opgenomen zijn die gecategoriseerd zijn volgens bepaalde grammaticale kenmerken. De zinnen worden gebruikt wanneer de leerling de werkwoordsvormen spelt met het algoritme. Om de leerlingantwoorden van informatieve feedback te kunnen voorzien moet het programma over kennis beschikken van grammaticale kenmerken van de in het bestand opgenomen zinnen. Hier valt bijvoorbeeld te denken aan de tijd waarin de zin staat, of er sprake is van een zwak of sterk werkwoord, de tijdsbepaling in de zin, wat het onderwerp is en of dat al dan niet in het enkelvoud staat enzovoort. Door nu in elke zin bepaalde codes mee te geven kan een daarvoor ontwikkelde procedure de benodigde informatie uit de zin halen. Tijdens het oefenen wordt steeds onmiddellijke feedback verschaft. Het programma geeft aan of een antwoord juist is of niet, maar ook waarom het goed of fout is. Hierdoor kan de leerling foutieve handelingen eenvoudiger opsporen en corrigeren. Bovendien biedt het algoritme het voordeel dat het programma exact kan bepalen op welk moment een verkeerde beslissing genomen is. Hoewel de keuze voor onmiddellijke feedback voor de hand ligt en goed beargumenteerd kan worden, komt uit de literatuur over computergestuurd onderwijs niet eenduidig naar voren of deze vorm van feedback in het algemeen het beste leerresultaat oplevert. Volgens Taylor (1987) moet

informatieve feedback (waarom is het antwoord goed of fout) uitgesteld worden tot het eind van een sequentie en op het moment gegeven worden wanneer de leerling opnieuw gaat oefenen. In de praktijk blijkt het merendeel van door hem beoordeelde COO-programma's hieraan niet te voldoen. Ook Rankin & Trepper (1978) vonden dat uitgestelde feedback effectiever was. Onmiddellijke feedback daarentegen is effectiever wanneer de leertaak betrekking heeft op kennisverwerving of wanneer de leerling moeilijkheden heeft met het beheersen van het materiaal (Cohen, 1985).

Aan welke condities feedbacksituaties in een gecomputeriseerde onderwijssituatie moeten voldoen om het leerproces optimaal te kunnen beïnvloeden, is dus niet geheel duidelijk. Via onderzoek proberen we hierover meer te weten te komen.

De aanbiedingsvolgorde van de leerstofonderdelen wordt bepaald door het programma. Hier is dus sprake van programmasturing. Op vele momenten vindt echter ook leerlingsturing plaats. De tijd die de leerling besteedt aan het lezen van een tekst of algemener: de wijze van verwerking van een scherm inhoud bepaalt hij of zij zelf. Daarnaast wordt bij elke oefening de gelegenheid gegeven om na een vastliggend minimum aantal extra items te oefenen of de leerweg te vervolgen. Verder kan de leerling op verschillende plaatsen in het programma een scherm terug gaan om een instructie nogmaals te lezen en om hulp of een voorbeeld vragen wanneer iets niet begrepen wordt. Aan het eind van elke les wordt een keuzemenu gepresenteerd met daarin de verschillende lesonderdelen die aan bod geweest zijn. De leerling kan één of meer onderdelen herhalen, waarbij de oefeningen met andere items gedaan worden. Een andere, meer ingrijpende vorm van leerlingsturing die in het programma gerealiseerd zal worden, is de mogelijkheid om op bepaalde momenten te kiezen voor een uitgebreide of beperkte instructie/feedback. Op deze manier kan de goede speller zich veel sneller door het programma werken.

Elk leerlingantwoord wordt door het programma geregistreerd en weggeschreven op een speciaal leerlingschijfje. Bij elke oefening moet de leerling aan een bepaald criterium voldoen om verder te kunnen. Lukt dat niet dan wordt nogmaals instructie gegeven, waarna nog een keer met andere items geoefend wordt. Gezien de omvang van het programma en de moeilijkheidsgraad van de leerstof zal de leerling gedurende een behoorlijk aantal sessies aan het programma werken. Het aantal sessies zal uiteraard afhangen van het kennisniveau, het leervermogen en het tempo van de leerling. In totaal zal het een leerling tussen de 15 en 30 uur kosten om de hele leergang door te werken, wat overigens slechts een fractie is van de tijd die hij/zij normaal aan deze leerstof besteedt. De tijdsduur van een sessie achter het toetsenbord mag willekeurig variëren, want het programma kan op elk moment beëindigd worden. Wanneer



dat gebeurt registreert het programma bij welke les en welk scherm de leerling gebleven is. Elke les is verdeeld in een aantal schermen die achter elkaar gepresenteerd worden. Een nieuwe leersessie wordt begonnen bij het scherm waar de vorige keer geëindigd is. In veel gevallen betekent dit een gedeeltelijke herhaling van een scherm. Deze "opfrissing" werkt uiteraard alleen maar positief.

#### 4. De eerste bevindingen

Voordat we ingaan op de eerste praktijkervaringen, geven we eerst aan hoe we in de uitprobeersfase te werk zijn gegaan. Nadat de eerste versie van het programma ontwikkeld is, begint het proces van uittesten en revisie. Dit uittesten dient zo systematisch mogelijk te geschieden. De doelgroep vormt hierbij de belangrijkste informatiebron. Om te kunnen zeggen hoe de leerling met het programma werkt en wat hij/zij ervan vindt moet de informatie zoveel mogelijk on-line verzameld worden. Wij onderscheiden drie manieren waarop dat gerealiseerd kan worden:

1. De leerlingresultaten worden door het programma geregistreerd. Hiermee komt een zeer gedetailleerd en omvangrijk databestand ter beschikking dat informatie bevat over het intypen van antwoorden, het maken van keuzes, het gebruiken van commando's, enzovoort.
2. Op bepaalde momenten in het programma vult de leerling een standaard vragenlijst in. Hiermee worden de ervaringen van de leerling met het programma geïnventariseerd.
3. De ontwikkelaar is zelf regelmatig aanwezig om leersessies achter het toetsenbord te observeren. Door het stellen van tussentijdse vragen en gerichte observatie kan hij aanvullende informatie verzamelen, bijvoorbeeld over het richten van de aandacht op het scherm en het toetsenbord en het gebruiken van commando's om bepaalde handelingen te kunnen uitvoeren.

Op het moment dat dit artikel geschreven wordt, hebben vier basisscholen (leerlingen uit groep 7 en 8) met de ontwikkelde lesmodulen geëxperimenteerd. In deze uitprobeersfase bleek het programma te voldoen aan de volgende doelstellingen:

1. Het programma moet zodanig opgezet zijn dat de leerling zich zelfstandig de leerstof eigen kan maken.
2. De technische bediening van de apparatuur en de programmatuur moet ook volledig aan de leerling kunnen overgelaten.
3. Het programma moet goed bestand zijn tegen allerlei onverwachte en zelfs gezochte antwoorden.
4. De leerkracht moet zoveel mogelijk ontlast worden.

Uit de vragenlijsten die de leerlingen tijdens het werken aan het programma hebben ingevuld, komt een positief oordeel naar voren. Alle vragen zijn terug te voeren op vijf clusters. Aan elk cluster ligt een voor ons belangrijke component voor de evaluatie ten grondslag. Het eerste cluster betreft de begrijpelijkheid van de instructie. Hierover gaven de leerlingen te kennen dat ze de uitleg in de meeste gevallen wel begrepen. Het tweede cluster gaat over de moeilijkheidsgraad van de lessen en oefeningen. De meeste leerlingen vonden de oefenstof niet zo moeilijk. Op de derde plaats waren we geïnteresseerd in de motivationele kant van het computer-gestuurde leerproces. Veel leerlingen werkten gemotiveerd en enthousiast aan het programma. De gemiddelde waardering kwam te liggen tussen "positief" en "zeer positief". Van een terugval in motivatie was na een paar maanden werken nauwelijks of geen sprake. De gemotiveerdheid van de leerlingen is hierdoor niet toe te schrijven aan het "nieuwheidseffect". Als vierde cluster namen we de begrijpelijkheid van de procedures die gevolgd moesten worden om met het programma te kunnen communiceren. Uit de resultaten bleek dat het merendeel van de leerlingen zonder al te veel problemen de gewenste handelingen kon verrichten. Wat betreft het bekend zijn van de gepresenteerde werkwoordsvormen (cluster 5) gaven de leerlingen aan de meeste te kennen. Hoewel we over diverse aspecten waardevolle gegevens hebben verzameld, kunnen we nog geen definitieve uitspraak doen over de effectiviteit van het programma. Ten eerste heeft dit te maken met het feit dat niet het gehele programma is uitgetest. Ten tweede is er een methodologisch probleem. De voorlopige indruk is dat er positieve leereffecten zijn, want de meeste leerlingen voldoen aan het criterium van 80% goede antwoorden dat bij alle oefenreeksen en testsessies gehaald moet worden. Hoe deze effecten precies moeten worden geïnterpreteerd is op dit moment echter niet eenduidig vast te stellen, want de leerlingen krijgen zowel van het COO-programma als van de leerkracht spellingles. Om de effectiviteitsvraag te kunnen beantwoorden is een onderzoek opgezet dat in het cursusjaar 1988/1989 uitgevoerd wordt.

Tenslotte nog enkele afsluitende opmerkingen. Het inrichten van gecomputeriseerde onderwijsleersituaties die tot doel hebben om leerlingen nieuwe kennis en vaardigheden bij te brengen is geen eenvoudige opgave. In feite moet de instructie zo zijn dat leerlingen met verschillende voorkennis en aanleg zich de leerstof eigen kunnen maken. Bij dat verwerkingsproces is het lezen van informatie op het beeldscherm van doorslaggevende betekenis. Daarbij gaat het niet zozeer om technisch lezen, maar vooral om begrijpend lezen. Hoewel we binnen het programma trachten de leesbaarheid en de duidelijkheid van de instructie optimaal te maken, blijft het de vraag in hoeverre de



aangeboden formuleringen toch problemen geven voor de zwakke lezers.

Verder vragen we ons af welke plaats de computer op wat langere termijn in het basisonderwijs zal veroveren. Het antwoord op deze vraag hangt samen met het probleem van de inpasbaarheid van de computer in het curriculum. In de uitprobeerfase viel het ons op dat de basisscholen zich allereerst als doel stelden "het omgaan met apparatuur en programmatuur". Van een werkelijke integratie in het curriculum was in deze fase geen sprake. Dit was overigens ook geenszins de bedoeling. Dat neemt niet weg dat het programma uiteindelijk in de plaats van de instructie door de leerkracht moet komen. Wanneer elke leerling één les van 45 minuten per week instructie per computer krijgt, betekent dit dat 30 leerlingen op één computer kunnen werken. Dat betekent ook dat het onderdeel "spelling" van het lesrooster zou kunnen verdwijnen. De vrij gekomen tijd kan aan andere onderwerpen besteed worden. Op de uitprobeerscholen wordt het programma nu naast het gewone taal/spellingprogramma gehanteerd.

De komende vijf jaar zal de hoeveelheid computers en software in het basisonderwijs aanzienlijk toenemen. Verschillende toepassingen zoals drill and practice, remediërend, verrijkend, tutorieel en andere zullen worden ontwikkeld voor diverse soorten leerstof. Sommige toepassingen zijn vooral geschikt voor speciale doelgroepen. Deze toename van mogelijke computertoepassingen zal een permanente integratie in het basisonderwijs op korte termijn zeker niet vereenvoudigen. Dit vereist naast het oplossen van veel praktische problemen, zoals het kunnen beschikken over voldoende apparatuur en de bijscholing van vele leerkrachten, vermoedelijk een ingrijpende bijstelling van het schoolwerkplan.

Utrecht, september 1988

## Bibliografie

- Assink, E.M.H., *Leerprocessen bij het spellen*. Aanzet voor de verbetering van de werkwoordsdidactiek. Dissertatie, Rijksuniversiteit Utrecht, 1983.
- Assink, E.M.H. & Klein, P., *De Werkwoordwinkel*. Purmerend, Muusses, 1984.
- Assink, E.M.H., De rol van grammaticale operaties bij het nemen van orthografische beslissingen. In: *Spektator*, 16/(3), 1987, 180-193.
- Bork, A., Nontrivial, Nonintelligent, Computer-Based Learning. In: *Contemporary Educational Psychology*, 12, 1987, 269-277.
- Cohen, V.B., A Reexamination of Feedback in Computer-Based Instruction: Implications for Instructional Design. In: *Educational Technology*, 25(1), 1985, 33-37.
- Daelemans, W., *Studies in Language Technology*. An Object-Oriented Computer Model of Morphophonological Aspects of Dutch. Dissertation, University of Leuven, 1987.
- Dirkzwager, A. & Mol, M., *Onderwijskundig computergebruik*. Amsterdam, Addison-Wesley Publishing Company, 1987.
- Geest, Th. van der, Computer-ondersteunend schrijfonderwijs: het ontwikkelen van een schrijffomgeving. In: *Spiegel*, 5-2(F), 1987, 77-93.
- Kattenberg, G.P.A., *Syntactische intuïtie en werkwoordsspelling op school*. Stageverslag Rijksuniversiteit Utrecht, Vakgroep Psychonomie, 1987.
- Landa, L.N., *Algorithmization in learning and instruction*. Englewood Cliffs, New Jersey: Educational Technology Publications, 1974.
- Pijls, F. & Kempen, G., Kennistechnologische leermiddelen in het grammatica- en spellingonderwijs. In: *Nederlands Tijdschrift voor de Psychologie*, 42, 1987, 354-363.
- Rankin, R. & Trepper, T., Retention and Delay of Feedback in a Computer-Assisted Instructional Task. In: *Journal of Experimental Education*, 46, 1978, 67-70.



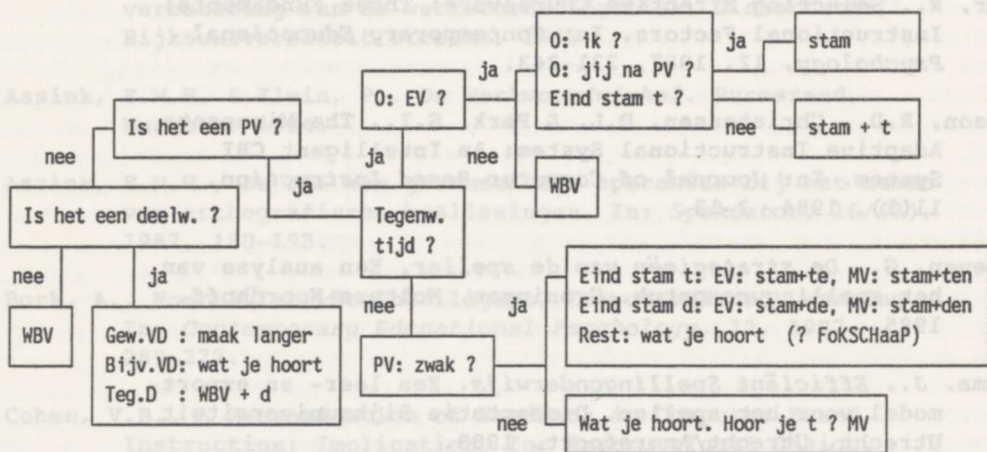
Reigeluth, C.M., *Instructional design theories and models*. An overview of their current status. Hillsdale, New Jersey Erlbaum, 1983.

Taylor, R., *Selecting Effective Courseware: Three Fundamental Instructional Factors*. In: *Contemporary Educational Psychology*, 12, 1987, 231-243.

Tennyson, R.D., Christensen, D.L. & Park, S.I., *The Minnesota Adaptive Instructional System: An Intelligent CBI System*. In: *Journal of Computer-Based Instruction*, 11(1), 1984, 2-13.

Verhoeven, G., *De strategieën van de speller*. Een analyse van het spellingvraagstuk. Groningen, Wolters-Noordhoff, 1985.

Zuidema, J., *Efficiënt Spellingonderwijs*. Een leer- en export-model voor het spellen. Dissertatie Rijksuniversiteit Utrecht. Utrecht/Amersfoort, 1988.



Het volledige algoritme waarmee in module 6 gewerkt wordt.

#### Afkortingenlijst:

WBV = Woordenboekvorm (infinitief)

Deelw. = Deelwoord

Gew.VD = Gewoon voltooid deelwoord

Bijv.VD = Bijvoeglijk gebruikt voltooid deelwoord

Teg.D = Tegenwoordig deelwoord

PV = Persoonsvorm

O = Onderwerp

EV = Enkelvoud

MV = Meervoud