

# COMPUTERS IN HET MOEDERTAALONDERWIJS

## Realiteit of science fiction ?

### Inleiding

*"De informatica draait op simpele siliciumchips, m.a.w. op zandkorrels waarmee een heel aparte wereld opgebouwd wordt, een wereld met een enorme impact op onze wereld of we dat wensen of niet. Ik ben ervan overtuigd dat we met deze zandkorrels ook ons voordeel kunnen doen in het moedertaalonderwijs. Maar dan moeten we dat op een verstandige manier doen. En voorlopig is het nog bijna uitsluitend science fiction."*

Met deze poëtische, maar vooral kritische bedenking besloot Daems (1985) zijn artikel in dit tijdschrift. Nu, meer dan twee jaar later, wil ik deze draad opnieuw opnemen. Ik wil allereerst de lezer waarschuwen : er is in tussentijd bitter weinig veranderd. De computerindustrie overspoelt de markt, waaronder ook de scholen, met de meest gesofisticeerde toestellen en levert daarbij 'handige' software om op een eenvoudige manier allerlei programma's te schrijven. Dit alles heeft echter voor de leerkracht Nederlands weinig opgeleverd : goede courseware voor het vak Nederlands is nog steeds zeer schaars.

Een oorzaak hiervoor is niet ver te zoeken : ondanks de grote opkomst van de computerlinguïstiek is men er nog steeds niet in geslaagd om het complexe gegeven taal in het enge keurslijf van een computerprogramma onder te brengen. Over de vorderingen hieromtrent is men evenmin erg optimistisch.

Moeten we dan geduldig afwachten tot er een 'intelligente' computer ontworpen wordt die op een natuurlijke wijze met zijn gebruiker kan communiceren ? Dat denk ik niet. Experimenten in Groot-Brittannië en de Verenigde Staten tonen aan dat er nu reeds 'schijnbaar' intelligente programma's kunnen ontworpen worden die op een zinvolle manier in het moedertaalonderwijs kunnen gebruikt worden. Het is een moeilijk en arbeidsintensief proces, maar het kan. Het blijkt bovendien de moeite om ermee door te gaan gezien de enthousiaste reacties van beide doelgroepen : leerkrachten en leerlingen.

Ik ben er mij wel van bewust dat er voor het Nederlands als moedertaal een bijkomende moeilijkheid is. De onderzoekscapaciteit voor ons kleine taalgebied is immers beduidend geringer. We staan nog helemaal aan het begin en er is nog heel wat onderzoek nodig om concrete resultaten te behalen. De vraag die steeds weer opduikt is : wie zal dat betalen ? Gezien de huidige besparingen zal deze vraag steeds prangender worden.

In dit artikel heb ik enkele concrete voorbeelden bijeengebracht van computerprogramma's die op een zinvolle wijze in bepaalde domeinen van het moedertaalonderwijs kunnen gebruikt worden. Dit in de hoop dat geïnteresseerden hierin inspiratie vinden en het als een uitdaging beschouwen om zelf aan de slag te gaan.

#### **BASIC, PILOT, ... welke software gebruiken voor COO ?**

Willen we de computer een reeks van opdrachten geven, m.a.w. willen we een computerprogramma schrijven, dan kan dit alleen via een gestandaardiseerde computertaal. Welke taal gebruiken we het best voor onderwijsprogramma's ? Hierover eerst enkele woorden uitleg.

#### **HOGERE PROGRAMMEERTALEN**

In de beginjaren van het Computerondersteund Onderwijs (in deze tekst afgekort tot COO) bestond er alleen een groot aantal hogere programmeertalen zoals BASIC, PASCAL enz. De toen ontwikkelde COO-programma's moesten uiteraard in deze talen worden geprogrammeerd. Al snel werd duidelijk dat dit voor een niet-informaticus aanzienlijke problemen met zich meebracht.

Eenvoudige drill-programma's kon een lesgever voor zijn leerlingen zelf programmeren. In het soort programma's waarin werd geprobeerd de student enige vrijheid in het formuleren van een antwoord te geven, ging het echter fout. Het is immers bijzonder moeilijk en het vergt veel tijd om in hogere programmeertalen fragmenten te schrijven die een antwoord van een leerling kunnen uiteenrafelen en zodanig 'begrijpen' dat er door het programma adequaat kan worden gereageerd.

Er groeide dan ook steeds meer de behoefte aan talen die in staat zouden zijn de lesgever bij het programmeren van de leersituatie een handje toe te steken.

Zo werden dan speciale auteurstalen voor COO ontwikkeld.

#### **AUTEURSTALEN EN AUTEURSSYSTEMEN**

Auteurstalen zijn zeer eenvoudige computertalen die een aantal specifiek voor COO ontwikkelde routines bevatten en aldus het programmeren heel wat makkelijker maken. Met eenvoudige commando's kan men een leerling de mogelijkheid geven om een min of meer vrij antwoord te geven door bepaalde toleranties bij het antwoorden in te bouwen : gebruik van synoniemen, extra woorden, hoofdletters, andere woordvolgorde, toelaten van spellingsfouten enz. Zonder veel moeite kan men verschillende juiste en foute antwoorden voorzien en aldus geïndividualiseerde feedback

inbouwen.

De meeste auteurstalen zijn ook voorzien van procedures voor het berekenen van scores, het opstellen van een inhoudstafel, het indelen van hoofdstukken, het opbouwen van helpmenu's, het voorzien van sprongen naar andere plaatsen in het programma enz.

De meest gebruikte auteurstalen zijn PILOT, COURSEWRITER, PLANIT en TENCORE.

Een andere mogelijkheid bieden de zogenaamde auteurssystemen. Dit zijn voorgeprogrammeerde systemen waarin 'raamwerken' opgenomen zijn, waarbinnen de auteur een aantal types van onderwijsprogramma's kan ontwerpen.

Met deze systemen kan een leerkracht zonder enige programmeerkennis moeiteloos een les programmeren.

Dit voordeel weegt naar mijn mening echter niet op tegen de nadelen van een auteurssysteem. De lessen die men ermee opbouwt hebben een vrij strakke structuur en de mogelijkheden die de auteur ter beschikking staan om variaties in zijn lessen aan te brengen, zijn vrij beperkt. Telkens men buiten het aangeboden standaardmodel wil treden, stuit men steeds op allerlei beperkingen.

Auteurstalen daarentegen zijn veel flexibeler : de auteur kan volledig vrij zijn scenario opbouwen zonder gebonden te zijn aan een bepaald standaardmodel.

Software die volledig geschikt is voor taalprogramma's bestaat nog niet. Onderstaande voorbeelden tonen echter aan dat er met de huidige auteurstalen toch al heel wat te verwezenlijken is.

#### **Enkele concrete voorbeelden**

##### **LEESONDERWIJS**

Ik kan me voorstellen dat het op het eerste gezicht niet duidelijk is wat men met een computer in het leesonderwijs kan aanvangen. Inderdaad, het langdurig lezen van een scherm is niet aan te raden. Een doodgewoon boek leest bovendien een stuk gemakkelijker en biedt allerlei mogelijkheden die een beeldscherm niet kan bieden : terugbladeren, aantekeningen maken, onderstrepen enz. Toch lijkt de computer een uitstekend medium om bepaalde aspecten van dit leesonderwijs te oefenen.

Een mooi voorbeeld hiervan is STORYBOARD. Ontworpen in Groot-Britannië, waarvan ook een variant in het Nederlands bestaat onder de naam Tekstcreatief (1).

Op het beeldscherm verschijnt een tekst, waar een leerling naar keuze langer of korter naar kan kijken. Daarna verdwijnt de tekst en ziet hij het beeldscherm gevuld met sterretjes die de grootte van de oorspronkelijke woorden

aangeven. Ook de leestekens blijven staan. De leerling moet de tekst reconstrueren door het raden van de woorden. Telkens hij een woord geraden heeft, wordt het door de computer op de juiste plaats gezet. De leerlingen beginnen meestal met het intypen van functiewoorden als de, het, van, en enz. en van woorden die zij zich menen te herinneren. Naarmate het spel vordert krijgen zij meer inzicht in de samenstelling van de tekst en gaan ze minder op hun geheugen werken. Op die manier sluit dit programma goed aan bij de training in 'voorspellend lezen' (Westhoff 1984).

Men kan nog een stap verder gaan door een tekst te laten reconstrueren die de leerling niet vooraf gezien heeft. Zijn enige houvast is dan de titel en de typografische structuur van de tekst.

Een voordeel van Storyboard is dat het hier niet gaat om een kant en klaar programma maar om een auteurssysteem. De leraar kan zelf zijn teksten kiezen in aansluiting op zijn lessen en deze in de computer intypen.

Op Storyboard bestaan trouwens tal van interessante varianten. Een voorbeeld, dat eigenlijk eerder thuishoort bij het schrijfonderwijs : leerlingen beluisteren een gesprek over een alledaags onderwerp in gewone spreektaal, compleet met herhalingen, tussenvoegsels, aarzelingen enz. Daarna krijgen ze de opdracht om dit informele gesprek om te zetten in geschreven taal door middel van Storyboard. De leerlingen kennen de context van de tekst al en zijn hier meer geconcentreerd op de formele aspecten van een geschreven tekst.

Wat opvalt bij dit soort programma's is de grote motivatie van de leerlingen. Ze beschouwen de computer als een tegen-speler waarvan ze het graag willen winnen.

Wanneer men in kleine groepjes werkt (3 tot 5) aan een computer, blijkt het dat de samenwerking sterk gestimuleerd wordt : het overleg dat ze moeten plegen om een volgende stap te zetten, leidt tot voortdurende discussie tussen de leerlingen.

TEXTEXPLORE, ontwikkeld aan de University of East Anglia in Norwich en thans nog in experimenteel stadium, is een project dat veel ingrijpender veranderingen teweeg brengt aan het leesonderwijs. Het is een programma dat expliciet aansluit bij een bepaald handboek. De makers gingen er vanuit dat zowel boek, papier, pen als computer even belangrijke hulpmiddelen kunnen zijn bij het werken met teksten.

De leerlingen vertrekken van gedrukte teksten en oefeningen uit het handboek. De computer wordt alleen dan gebruikt waar hij werkelijk iets toevoegt aan het leerproces.

Figuur 1 geeft een overzicht van het programma. Het kan op twee manieren gebruikt worden : meestal begint men met pad A dat allerlei oefeningen bevat die betrekking hebben op een bepaalde tekst. Naar eigen goeddunken kan de leerkracht overschakelen naar pad B om specifieke

vaardigheden verder te oefenen.

Textexplore is een zeer flexibel programma : op ieder moment kan naar een ander niveau overgestapt worden. Het bevat ook een grote variatie van oefeningen die erop gericht zijn het geheel van leesvaardigheden te ontwikkelen.

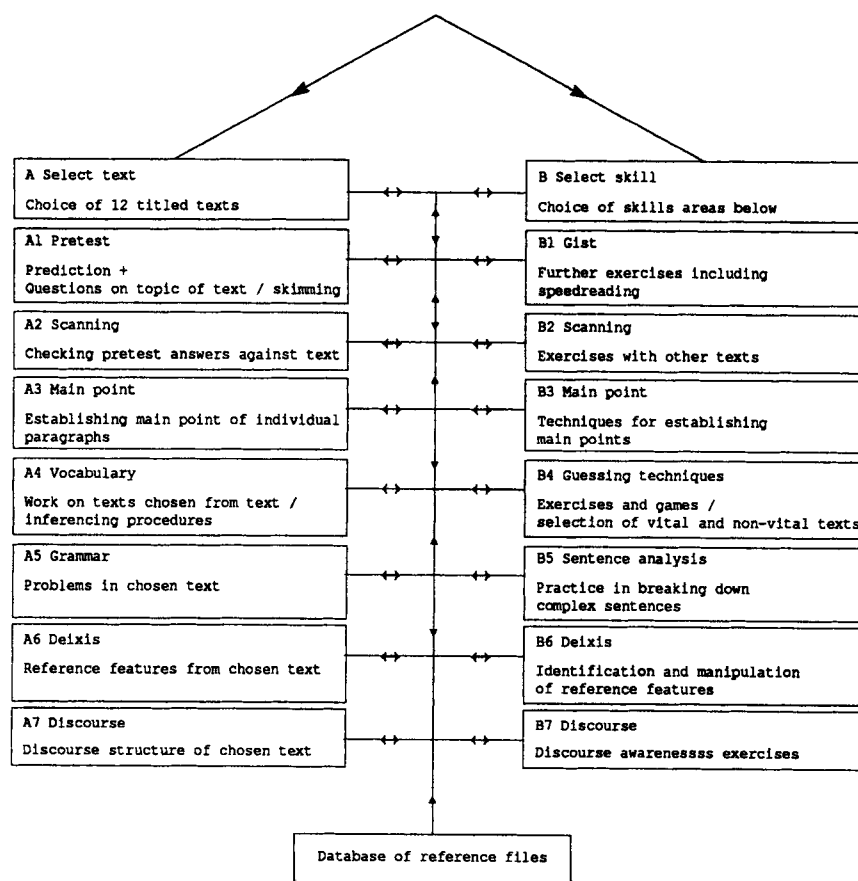


Fig. 1. : Textexplore (Clarke, 1986)

Verder uitweiden over Textexplore zou ons te ver leiden. Ik vermeld het hier vooral omdat het een uitstekend voorbeeld is van hoe een computer als specifiek medium in een domein van het taalvaardigheidsonderwijs geïntegreerd is, naast het handboek.

Het realiseren van zo'n project voor de les Nederlands zou heel mooi zijn, maar misschien irrealistisch. Het gaat hier immers om het ontwerpen van een didactisch verantwoord geheel waarin handboek en computeroefeningen volledig bij elkaar aansluiten, iets wat al vlg honderden uren arbeid vraagt.

De technische realisatie van dergelijke computeroefeningen

vormt niet zo'n probleem. Het is een typisch voorbeeld van een programma dat zeer goed in een auteurs taal kan geprogrammeerd worden.

#### SPELLING- EN SPRAAKKUNSTOEFENINGEN

De eerste gedachte die bij vele leerkrachten opkomt wanneer wordt gesproken over computers bij het talenonderwijs is 'drill- en practice' oefeningen. In onderwijskringen wordt hierover de laatste tijd erg minachtend gesproken, ten onrechte naar mijn mening. 'Goede' drilloefeningen kunnen een uitstekend middel zijn voor het verwerven van automatisen bij allerlei spellings- en grammatiecamoeilijkheden zoals werkwoordspelling en zinsontleding. De computer is immers een uiterst geduldige oefenmeester die een leerling steeds andere oefeningen over een bepaald onderwerp kan voorschotelen en geïndividualiseerde feedback kan voorzien.

Wat verstaat men echter onder 'goede' drilloefeningen? Het programmeren van zo'n oefening vormt in een auteurs taal weinig problemen. Wat veel belangrijker en bovendien veel moeilijker te realiseren is, is de didactische onderbouw van een drillprogramma.

Allereerst moet men een grondige analyse maken van de opeenvolgende denkstappen die een leerling moet doorlopen om tot een juiste oplossing te komen. Pas dan kan men verschillende denkfouten onderscheiden en hierop zeer gerichte feedback geven. Men kan dan ook een duidelijke foutenanalyse opstellen die niet alleen vertelt hoeveel fouten de leerling gemaakt heeft, maar ook welke denkstappen hij nog niet onder de knie heeft en op die manier een uitstekend hulpmiddel zijn bij remedial teaching.

Een uitstekend voorbeeld hiervan is de cursus EIGENWIJZER, een leer- en oefenprogramma voor werkwoordspelling (2). Zoals Textexplore is ook Eigenwijzer geen geïsoleerd computerprogramma. Het maakt deel uit van een taalproject dat gebruik maakt van verschillende middelen: bord, papier en pen, handboek, overheadprojector en computer.

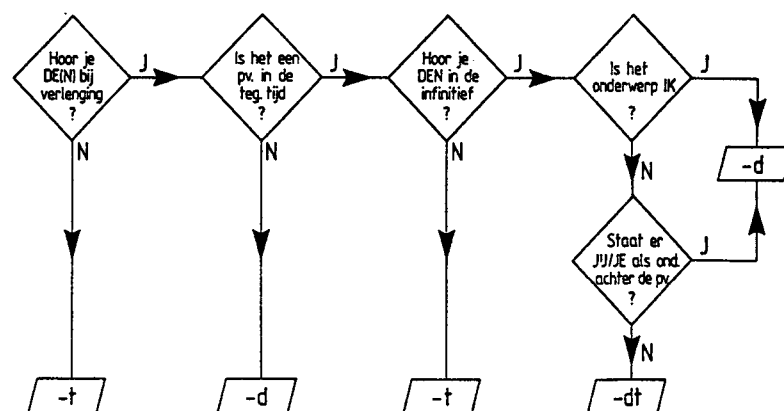
Het programma verloopt in verscheidene fasen.

Allereerst krijgt de leerling een geheel van denkstappen aangeboden om tot een juiste werkwoordspelling te komen (Pepermans & Van Herck 1985). Vervolgens leert hij met dit algoritme om te gaan en te memorizeren.

Het bijhorende computerprogramma kan in beide stadia gebruikt worden.

In het leerprogramma worden met behulp van een vragen-antwoordspel de verschillende denkstappen aangeleerd (fig. 2).

Het oefenprogramma geeft de leerling vervolgens de kans om zijn kennis te toetsen. In willekeurige zinnen moet hij de werkwoorduitgang invullen. Maakt hij een typefout, dan krijgt hij een nieuwe kans. Wanneer hij een fout tegen



© 1985 by G. de Schutter.  
Bijlage bij: J. PEPERMANS & P. VAN HERCK, *Eigenwijzer. Een nieuwe methode voor werkwoordspelling*. Leuven/Amersfoort, 1985.

## **EIGENWIJZER 2** : je hoort -TE(N) of -DE(N)

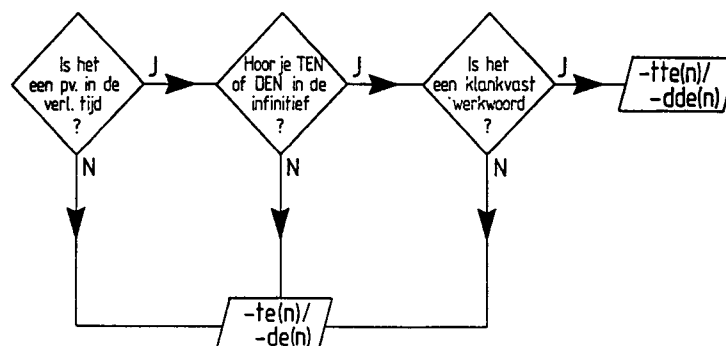


Fig. 2.

de werkwoordspelling maakt, kan hij teruggrijpen naar de vragen in het leerprogramma en zodoende zelf ontdekken welke verkeerde denkstap hij gemaakt heeft. Vervolgens krijgt hij een zin aangeboden met hetzelfde type van werkwoordsvorm waartegen hij een fout gemaakt heeft. Aldus wordt het programma automatisch aangepast aan de specifieke behoefte van de leerling.

Ook de feedback is hier sterk geïndividualiseerd. Hij bestaat niet uit voorgeprogrammeerde zinnen, maar uit het hernemen van de verschillende denkstappen al naargelang de fout die de leerling gemaakt heeft.

Een veelbelovend domein voor de linguïstiek en naar mijn mening ook voor het grammaticaonderwijs is de artificiële intelligentie. Hier probeert men regels te formuleren in een programmeertaal om via een computer natuurlijke taal te beschrijven en te genereren. Deze discipline staat weliswaar nog in zijn kinderschoenen, maar zal in de toekomst wellicht ook voor het onderwijs veel mogelijkheden bieden.

Een programmeertaal die zich uitstekend voor taalbeschrijving leent is PROLOG (Clocksin & Mellish 1981).

Een Prologprogramma bestaat uit een geheel van regels en data waarmee de computer automatisch allerlei mogelijke modellen genereert of een ingetikt model op zijn juistheid controleert.

Het COO-programma FROG (French Robust Grammar Checking) bewijst dat Prolog ook voor het grammatica- en schrijfonderwijs bruikbaar kan zijn (Imlah & Boulay, 1985). Het programma bestaat uit een geheel van grammaticaregels en een databank van woorden op basis waarvan automatisch juiste zinnen kunnen gegenereerd worden.

De leerling tikt een willekeurige zin in de computer. Het programma toetst deze zin aan de geformuleerde regels en de databank en deelt mee welke fouten gemaakt worden (fig. 3).

Een Prologprogramma dat elke willekeurige Nederlandse zin op lexicale, grammaticale en spellingsfouten controleert is in de nabije toekomst zeker niet realiseerbaar. Wel denk ik dat PROLOG een zeer geschikte taal is om bepaalde deelaspecten uit de grammatica en spelling te oefenen. Zoals bij de drill oefeningen is ook hier weer het ontwerpen van een didactische onderbouw de allereerste vereiste: nl. het formuleren van duidelijke regels die de leerling moet toepassen.

**> Le bateaus est sorte/e par mon frere qui portira la pain pour le mangir.**

**Le bateaus**

Unrecognized word „bateaus”.  
Assumed to be noun (m,p).

Determiner le (m,s) does not agree with noun bateaus (m,p).  
Should be „les”.

**est**

Verb „est” does not agree with subject „bateaus”.

**sorte/e**

Unrecognized word „sorte/e”.  
Assumed to be -ir type (f,s) verb past participle.  
Should be „... sortie ...”

Past participle sorte/e (f,s) does not agree with subject bateaus (m,p). Should be „sortis”.

**par mon frere**

Unrecognized word „frere”.  
Assumed to be noun (m,s).



qui portira  
Unrecognized word „portira”.  
Assumed to be a verb, future tense, (third, s).

La pain

Determiner la (f,s) does not agree with noun pain (m,s).  
Should be „le”.

pour le manglr

Unrecognized word „manglr”.  
Assumed to be -ger type verb infinitive „manger”.

Fig. 3 : FROG (Inlah, 1985)

### Besluit

Bovenstaande voorbeelden vormen beslist geen volledig overzicht van wat er met een computer mogelijk is in het moedertaalonderwijs. Zo vermeld ik bijvoorbeeld niets over het gebruik van tekstverwerkingssystemen bij het schrijfonderwijs, een recentelijk opkomend fenomeen dat in dit tijdschrift reeds uitgebreid aan de orde was (Valcke & Yde 1986 en Cuvelier 1987).

Ik wil hiermee ook niet beweren dat de computer in het moedertaalonderwijs 'moet' gebruikt worden, maar wel dat het een nuttig hulpmiddel kan zijn bij bepaalde domeinen van het moedertaalonderwijs. Een bijkomende vereiste is dat men over goede programma's beschikt. Dit is juist de grote moeilijkheid voor ons taalgebied : belangstelling van de kant van de leerkrachten is er volop maar wildgroei bestaat er ook in grote mate. Bij de overheid ligt wellicht de taak om afzonderlijke initiatieven te coördineren en te begeleiden. Gezien de huidige besparingen zie ik deze toekomst echter somber in.

Karla De Ceulener, Bafferstraat 5, 2800 Mechelen

### NOTEN

- (1) TEKSTCREATIEF wordt verdeeld door Wida Software, Londen.
- (2) EIGENWIJZER wordt verdeeld door Uitgeverij Plantijn, Antwerpen.

### BIBLIOGRAFIE

B. Camstra, Leren en onderwijzen met de computer. Stenfert Kroese, Leiden, 1980.

D.F. Clarke, Computer-assisted reading. What can the machine really contribute ? System, 14/1, 1-13, 1986.

- W. Clocksin & C.S. Mellish, Programming in PROLOG. Springer-Verlag, 1981.
- P. Cuvelier, Tekstverwerken in de schrijffles. Vonk, 17/1, 36-49, 1987.
- Fr. Daems, Een wereld in een zandkorrel. Computers in het moedertaalonderwijs. Vonk, 15/1, 1-13, 1985.
- Fr. Daems, Kaf onder het koren. Over didactisch (on)verantwoorde programmatuur voor het moedertaalonderwijs. in : W. Decoo, P. Reynders & H. Sevenans (reds.). Computers & Onderwijs. Een symposiumverslag over informatica, talen, wetenschappen en lager onderwijs ; Antwerpen, CBL-UIA, 47-58.
- J. Higgins & T. Johns, Computers in language teaching. Collins, London, 1984.
- W.G. Imlah & J.B.H. du Boulay, Robust natural language parsing in CAI. System, 13/2, 137-147, 1985.
- P. Looijmans & D. Schrauwen, De docerende computer en andere sprookjes. Moer, 1986/6, 2-8.
- J. Pepermans & P. Van Herck, Eigenwijzer. Nieuwe methode voor werkwoordspelling. (Deel voor de leerling : toelichting). Acco, Leuven, 1985.
- M.Valcke & P. Yde, Tekstverwerking op school : een toverstok voor het schrijfvaardigheidsonderwijs ? Vonk, 16/1, 20-27, 1986.
- G.J. Westhoff, Voorspellend lezen. Wolters-Noordhoff, Groningen, 1981.