

Pragmatisch redeneren in model-gebaseerde diagnose¹

R.R. Bakker
M. Bourseau
Universiteit Twente

Samenvatting

Model-gebaseerde diagnose bepaalt de defecte componenten in een technisch systeem door middel van een gerichte reeks testen en metingen. Meetadvies wordt gegeven op basis van verkregen diagnostische hypothesen. De huidige model-gebaseerde diagnostische methoden² hebben een hoge berekeningscomplexiteit (in $O(2^n)$) waarbij n het aantal componenten in het model is). De Pragmatic Diagnostic Engine (PDE) is een model-gebaseerde redeneermethode die een polynomiale berekeningscomplexiteit heeft (in $O(n^2)$). Door een beperking van het diagnostisch redeneren tot de belangrijkste diagnostische hypothesen, kan in korte tijd een goed meetadvies gegeven worden. In een aantal experimenten werd bevestigd dat PDE's meetadvies nagenoeg hetzelfde is als dat van de *standaardmethode* GDE.

Trefwoorden

Model-gebaseerde diagnose, redeneren

1 Inleiding

Model-gebaseerde diagnostische methoden zijn gericht op de lokalisering van defecten in een technisch systeem. Met behulp van een simulatiemodel van het systeem en observaties worden diagnostische hypothesen gegenereerd. Meestal zijn er meerdere verklaringen mogelijk voor het afwijkende gedrag van het systeem. Aanvullende metingen of tests zijn dan noodzakelijk. (Davis, 1988) is een inleiding op model-gebaseerde diagnose. (Hamscher, 1992) bevat vaak geciteerde publicaties op het gebied van model-gebaseerde diagnose.

GDE

Een standaard methode voor diagnostisch redeneren is GDE (General Diagnostic Engine (De Kleer, 1987)). GDE gebruikt een model en waarnemingen om *conflicten*³ te berekenen. Deze conflicten worden omgezet in *diagnosen*⁴. Ten slotte worden deze diagnosen gebruikt om een goed discriminerend meetpunt te berekenen. Het proces van waarnemen, diagnostiseren en meetadvies wordt herhaald tot de defecten bekend zijn. Figuur 1 beschrijft de architectuur van GDE.

GDE's berekeningscomplexiteit is hoog. Het bepalen van (*minimale*)⁵ conflicten en diagnose is in $O(2^n)$ (n is het aantal componenten in het model) in eindige domeinen

¹ Dit onderzoek is verricht in het kader van project SKBS-A1. De Stichting Knowledge-Based Systems stimuleert de samenwerking tussen universiteiten en het bedrijfsleven op het gebied van de kennistechnologie.

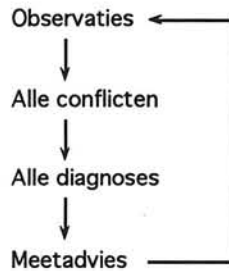
² Tenzij vereenvoudigende aannames worden gemaakt.

³ Een conflict is een verzameling van componenten waarvan er tenminste één defect moet zijn. Voor formele definities, zie bijvoorbeeld (De Kleer, 1992).

⁴ Een diagnose is een verzameling van mogelijk defecte componenten.

⁵ Een conflict (diagnose) is minimaal als geen strikte deelverzameling een conflict (diagnose) is.

zoals digitale elektronica, en onbeslisbaar in oneindige domeinen zoals analoge elektronica. In figuur 2 is de bepaling van conflict <C1> onbeslisbaar.



Figuur 1. Architectuur van GDE

Verder kan de oplossingsgrootte (het aantal conflicten en diagnoses) te groot zijn (opnieuw in $O(2^n)$). Figuur 5, beschrijft een worst-case voorbeeld. De berekeningsproblemen in GDE zijn ook in de praktijk zo groot dat GDE alleen toegepast kan worden in kleine systemen.

$$\begin{array}{ccccc}
 & A & & B & & C \\
 10 & \rightarrow & C_1: B = A & \rightarrow & C_2: C = B^5 + B^4 + B^3 + B^2 + B & \rightarrow & 33
 \end{array}$$

Figuur 2. Bepaling van conflict <C1> is onbeslisbaar bij geheeltallige domeinen

Focused Sherlock

Focused Sherlock (De Kleer, 1991) is de voor deze bijdrage belangrijkste verbetering van GDE. Focused Sherlock genereert niet alle mogelijke diagnoses, maar beperkt de berekeningen tot de meest waarschijnlijke. Verder worden ook alleen 'noodzakelijke' conflicten bepaald. In figuur 3 wordt de architectuur van focused Sherlock beschreven. Op grond van waarnemingen en al verkregen conflicten wordt de meest waarschijnlijke *deeldiagnose*⁶ gegenereerd. Deze deeldiagnose wordt gecontroleerd. De controle van een deeldiagnose kan al dan niet tot leiden tot nieuwe conflicten. In het eerste geval is de deeldiagnose een diagnose. De generatie van diagnoses wordt voortgezet tot aan een stopcriterium voldaan wordt (zoals de 5 meest waarschijnlijke diagnoses zijn bepaald). Daarna wordt meetadvies gegeven gebaseerd op de berekende diagnoses.

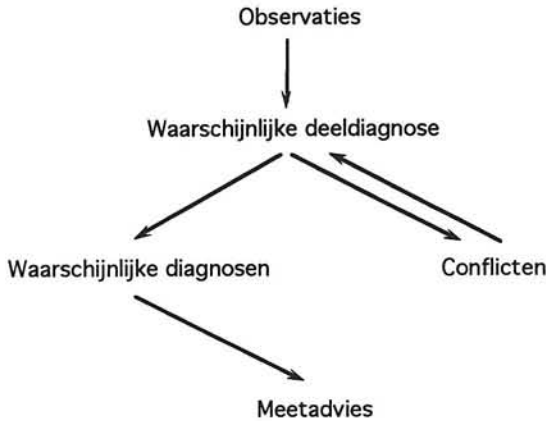
De worst-case berekeningscomplexiteit van focused Sherlock is tenminste in $O(2^n)$ omdat de consistentiecontrole van een deeldiagnose een NP-volledig probleem is in eindige domeinen. We weten niet of de hoge worst-case complexiteit praktische toepassingen verhindert. De hoge worst-case complexiteit is wel onze motivatie geweest om te onderzoeken of polynomiale methoden voor meetadvies mogelijk zijn.

Overzicht

Aangezien de worst-case complexiteit van het bepalen van een diagnose in $O(2^n)$ is, hebben we onderzocht of meetadvies gebaseerd kan worden op een kleine verzame-

⁶ Een deeldiagnose is een verzameling van mogelijk defecte componenten die de tot dan toe gevonden conflicten dekt.

ling van berekenbare conflicten. In §2 wordt de resulterende methode, PDE (Pragmatic Diagnostic Engine) beschreven. In §3 gaan we in op experimentele resultaten die erop duiden dat PDE's en GDE's meetadviezen van gelijke kwaliteit zijn. Tot slot gaan we in §4 in op een uitbreiding van PDE die meetadvies verbetert als er foutmodellen van componenten in het model beschikbaar zijn.



Figuur 3. Architectuur van focused Sherlock

2 Pragmatic Diagnostic Engine (PDE)

Aangezien de worst-case complexiteit van het bepalen van een diagnose tenminste in $O(2^n)$ is, hebben we onderzocht of meetadvies gebaseerd kan worden op een kleine verzameling van berekenbare conflicten. Eerst zullen we beschrijven hoe meetadvies gegeven kan worden op grond van conflicten. Dit lost de computationele problemen nog niet op, omdat (1) het aantal conflicten in $O(2^n)$ is, (2) het bepalen van een conflict onbeslisbaar kan zijn en (3) de berekening van het beste meetpunt een nieuwe bron van exponentialiteit bevat. Daarom beschrijven in §2.2 hoe meetadvies gegeven kan worden op basis van *enkelvoudige* en *tweevoudige* conflicten. De resulterende methode heet PDE, de Pragmatic Diagnostic Engine.

2.1 Meetadvies op basis van conflicten

Meetadvies op basis van conflicten bouwt voort op meetadvies op grond van diagnosen van De Kleer en Williams (1987). De a priori faalkans van componenten moet bekend zijn, en we nemen verder aan dat componenten onderling onafhankelijk falen. Voor elk signaal x_i kunnen mogelijke waarden v_{ik} worden gemeten. De verwachte informatiewinst ΔH van meetpunt x_i is:

$$\Delta H = \sum_k p(x_i = v_{ik}) \log(p(x_i = v_{ik})) + \xi$$

In deze formule is ξ een restterm die verwaarloosd mag worden in de context van deze bijdrage; $p(x_i = v_{ik})$ is de voorwaardelijke kans dat $x_i = v_{ik}$ gegeven de huidige

verzameling van diagnoses. Het meetpunt met minimum ΔH discrimineert het best tussen de meest waarschijnlijke diagnoses.

De kansen $p(x_i = v_{ik})$ kunnen ook berekend worden met behulp van de conflicten. We moeten eerst enkele begrippen introduceren. Laat $C = \{c_1, \dots, c_l\}$ een conflict zijn; $P_{\text{prior}}(c_i)$ is de a priori faalkans van component c_i . De a priori kansmassa $P(C)$ van uitgesloten toestandstoewijzingen (normaal/abnormaal) gegeven een conflict C is:

$$P(C) = \prod_{i=1}^l (1 - P_{\text{prior}}(c_i))$$

De a priori kansmassa $P_S(C)$ van uitgesloten toestandwijzigingen gegeven een verzameling van conflicten $\mathbf{C} = \{C_1, \dots, C_l\}$ is:

$$P_S(C) = \sum_{C \in \mathbf{C}} P(C) - \sum_{C_i, C_j \in \mathbf{C}, i < j} P(C_i \cup C_j) \\ \dots (-1)^{m+1} P(C_1 \cup \dots \cup C_m)$$

De a priori kansmassa van alle diagnoses gegeven een verzameling van conflicten C is $1 - P_S(\mathbf{C})$. De kans $p(x_i = v_{ik})$ kan nu berekend worden als de verzameling resterende conflicten \mathbf{C}_k na meting van $x_i = v_{ik}$ bekend is. In dat geval berekenen we $P_{ik} = 1 - P_S(\mathbf{C}_k)$, de a priori kansmassa is van de resterende diagnoses na meting van $x_i = v_{ik}$. Nu volgt:

$$p(x_i = v_{ik}) = P_{ik} / \left(\sum_j P_{ij} \right)$$

De verzameling van resterende conflicten kan berekend worden met behulp van de al verkregen dependency records. Meetadvies kan dus verkregen worden op basis van conflicten; alleen de berekeningscomplexiteit is erg hoog. Alle conflicten zijn vereist, de resterende conflicten na meting moeten bepaald worden en de berekening van $p(x_i = v_{ik})$ is exponentieel in het aantal conflicten.

2.2 Meetadvies op basis van n-voudige conflicten

Een conflict treedt op als een uitvoer van een component verschillende waarden (berekend of waargenomen) aan kan nemen. Dit kan alleen optreden als een of meer waargenomen uitgangen van het systeem gebruikt zijn in de berekeningen of vergelijkingen. Een *enkelvoudig conflict* is een conflict dat een uitgang gebruikt. Enkelvoudige conflicten kunnen bepaald worden door simulatie van het systeem (voorwaartse propagatie) en vergelijking van berekende en waargenomen uitgangen. Enkelvoudige conflicten kunnen dus altijd simpel berekend worden, waardoor beslisbaarheidsproblemen niet optreden.

Een *tweevoudig* conflict is een conflict waarbij hoogstens twee uitgangen gebruikt zijn. De meeste tweevoudige conflicten kunnen berekend worden door simulatie van het systeem gevolgd door een achterwaartse propagatie van waarden in het systeem. Achterwaartse propagatie is niet altijd eenvoudig, zie bijvoorbeeld figuur 2. We beperken daarom de achterwaartse propagatie tot die componenten in het model met expliciete achterwaartse propagatieregels. Hierdoor is het beslisbaarheidsprobleem opnieuw verdwenen.

Beperkte propagatie kan gegeneraliseerd worden tot m-voudige conflicten die m uitgangen gebruiken. Deze conflicten worden bepaald door (m-1) keer te wisselen tussen voorwaartse en achterwaartse waardenpropagatie. Als een systeem k uitgangen heeft bestaan er hoogstens k-voudige conflicten.

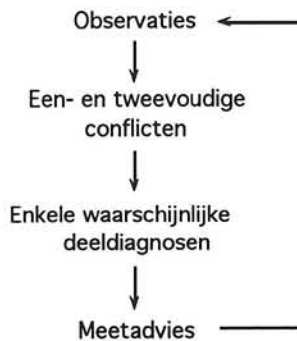
De a priori kansmassa $1 - P_s(\mathbf{C})$ van uitgesloten diagnoses gegeven de verzameling van conflicten kan geschat worden op grond van de verzameling van enkelvoudige en tweevoudige conflicten \mathbf{C}_2 . De relatieve fout $[P_s(\mathbf{C}) - P_s(\mathbf{C}_2)]/[1 - P_s(\mathbf{C})]$ is theoretisch moeilijk te bepalen. In experimenten (zie ook §3 bleek deze fout klein ($< 1\%$) te zijn. Enkelvoudige conflicten blijken informatief te zijn; ze zijn direct gerelateerd aan afwijkende uitgangen. Tweevoudige conflicten elimineren simpele verklaringen voor twee afwijkende uitgangen of een afwijkende en een goede uitgang. De n-voudige conflicten dekken meer gecompliceerde situaties. De beperking tot een- en tweevoudige conflicten is zelfs nog niet voldoende om combinatorische explosies uit te sluiten.

In het geconstrueerde voorbeeld in figuur 5, bestaat een exponentieel aantal enkelvoudige conflicten. We hebben dit probleem opgelost door een beperking van het aantal dependency records van een componentuitgang. De keuze welke dependency records bewaard blijven wordt lokaal gemaakt door de twee sterkst-verschillende op te slaan. Als meerdere records evenveel van elkaar verschillen worden er twee willekeurig gekozen. In onze experimenten bleek dat twee dependency records per propagatie voldoet in de zin dat de kwaliteit van het meetadvies nagenoeg hetzelfde blijft.

Approximatie van $p(x_i = v_{ik})$

Het resterende probleem is de berekening van $p(x_i = v_{ik})$. Een simpele oplossing is om een aantal meest waarschijnlijke *deeldiagnosen* te genereren en deze kans te berekenen op dezelfde wijze als in focused Sherlock die kans berekend wordt op grond van diagnoses.

Een aanpak die De Weger (1992) gebruikte, is om een ondergrens en bovengrens voor $1 - P_s(\mathbf{C}_{lim})$ bij te houden en de berekening te stoppen als het relatieve verschil kleiner is dan een grenswaarde (bijvoorbeeld 0.05).



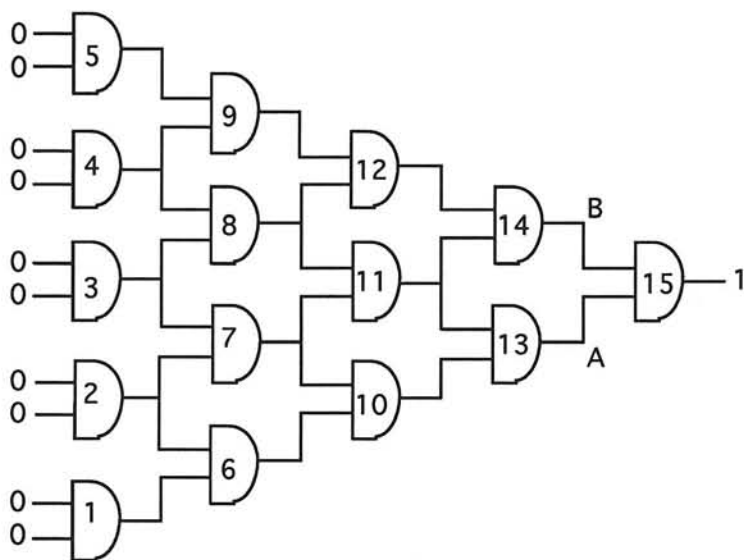
Figuur 4. Architectuur van PDE

Pragmatic Diagnostic Engine

PDE, de pragmatic diagnostic engine, bepaalt dus een beperkte verzameling van een- en tweevoudige conflicten op grond van de waarnemingen, en geeft meetadvies gebaseerd op het discriminerend vermogen van de meetpunten. De architectuur van PDE is weergegeven in figuur 4.

We nemen het model in figuur 5 als voorbeeld. PDE berekent hier twee enkelvoudige conflicten door simulatie, bijvoorbeeld: $\langle n(1), n(6), n(10), n(13), n(15) \rangle^7$ en $\langle n(2), n(7), n(11), n(14), n(15) \rangle$.

Het systeem heeft maar een uitgang, zodat alleen enkelvoudige conflicten gegenereerd worden. De vijf meest waarschijnlijke deeldiagnosen zijn $[ab(15)]$, $[ab(1), ab(2)]$, $[ab(1), ab(7)]$, $[ab(1), ab(11)]$, en $[ab(1), ab(14)]$. Van deze deeldiagnosen is alleen $[ab(15)]$ een diagnose; de andere deeldiagnosen dekken slechts de tot nu bepaalde conflicten. Volgens PDE's meetadvies zijn signalen A en B de meest informatieve metingen. Na meting $A = 0$ is het overblijvende conflict $\langle n(15) \rangle$; de correcte diagnose $[ab(15)]$ is dus bepaald. In dit voorbeeld bepaalt PDE de echte defecte componenten na evenveel aanvullende metingen (één!) als GDE. PDE berekent drie conflicten en geen diagnosen, terwijl GDE 17 conflicten en 33 diagnosen berekent.



Figuur 5. Alle componenten zijn and-poorten. Component 15 is defect

3 Experimentele resultaten

We hebben PDE getest op een aantal voorbeelden zoals n-bit full adders en het bindec-systeem een binary-decimal converter die 30 componenten bevat. PDE is geïmplemen-

⁷ Conflicten worden weergegeven door $\langle \rangle$, diagnosen door $[\]$; $n(1)$ betekent dat component 1 normaal verondersteld wordt; $ab(1)$ betekent dat component 1 defect verondersteld wordt.

teerd in Quintus Prolog op een Sun sparc station 1; we hebben geen pogingen tot runtime optimalisatie ondernomen.

Experimenten met enkelvoudige en dubbele fouten in een 100-bit adder werden gelokaliseerd na 2-3 aanvullende metingen voor enkelvoudige fouten en 4-6 aanvullende metingen voor dubbele fouten. De vereiste CPU-tijd varieerde tussen de 5 en 20 seconden. De fout gemaakt door de beperking tot een- en tweevoudige conflicten was te verwaarlozen. Het aantal aanvullende metingen is gelijk aan dat van focused Sherlock. Runtime performance van Sherlock (op een Symbolics) is beter (enkele seconden). We hebben met enkelvoudige fouten in het bindec-systeem geëxperimenteerd. Deze werden na 3-5 aanvullende metingen bepaald; alle fouten werden binnen 20 seconden CPU-tijd gelokaliseerd. Een vergelijking tussen PDE en GDE was alleen mogelijk voor kleine systemen. Voor grotere systemen moeten GDE's berekeningen wegens overmatige rekentijden afgekap worden. Voor de kleine systemen geldt dat meetadvies in beide systemen identiek is, alleen PDE is aanzienlijk sneller.

De resultaten van PDE worden ook bevestigd door de tijdsafhankelijke variant PDE-T die dynamische systemen met feed-back diagnostiseert (De Weger, 1992). In een serie experimenten, bleek dat de beperking tot enkel- en tweevoudige conflicten de berekeningsproblemen aanzienlijk reduceert, terwijl de kwaliteit van het meetadvies nagevoeg hetzelfde blijft.

4 Foutmodellen

Foutmodellen beschrijven het afwijkend gedrag van componenten. Voor model-gebaseerde diagnose is de beschikbaarheid van deze modellen niet vereist. Als foutmodellen beschikbaar zijn, kunnen ze gebruikt worden om de diagnosen nauwkeuriger te bepalen. Foutmodellen in de context van model-gebaseerde diagnose zijn het eerst beschreven door De Kleer (1989) en Struss (1989). In deze paragraaf gaan we ervan uit dat foutmodellen het afwijkende gedrag van een component volledig beschrijven (bijvoorbeeld het vermogen van een defecte motor is 0); we noemen dit *expliciete foutmodellen*. Impliciete foutmodellen (bijvoorbeeld, het vermogen van een defecte motor is minder dan het normale vermogen) laten we buiten beschouwing.

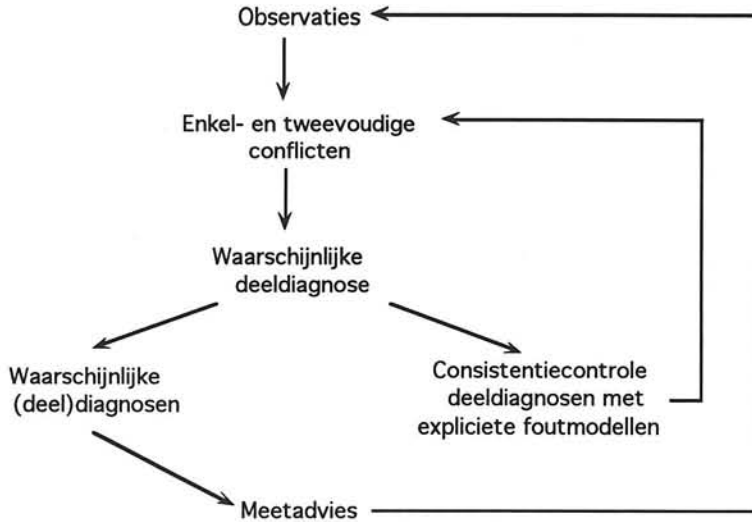
Naast expliciete foutmodellen veronderstellen we dat elke component een onbekende fouttoestand kent. Hierdoor kunnen foutmodellen gebruikt worden om tot een betere ordening te komen van de diagnostische hypothesen. Een groot voordeel van expliciete foutmodellen is dat deeldiagnosen die bestaan uit expliciete foutmodellen direct door simulatie op consistentie getest kunnen worden. Er zijn dus geen berekeningsproblemen voor het toetsen van deze klasse van deeldiagnosen. We hebben daarom PDE aangepast om deeldiagnosen te kunnen toetsen die alleen expliciete foutmodellen bevatten. De resulterende architectuur is weergegeven in figuur 6.

Experimenten zullen moeten aantonen in hoeverre foutmodellen het diagnostisch proces versnellen. Voor zover wij weten zijn, na de theoretische introductie van foutmodellen, deze experimenten nog niet systematisch uitgevoerd.

5 Conclusies

In deze bijdrage is een model-gebaseerde diagnostische methode PDE beschreven met polynomiale berekeningcomplexiteit, terwijl het meetadvies van even goede kwaliteit is als GDE's meetadvies.

Experimentele resultaten duiden erop dat PDE defecte componenten in een korte tijdsperiode opspoorde. PDE laat zich eenvoudig uitbreiden naar het gebruik van expliciete foutmodellen. Aanvullende experimenten zijn vereist om een beter inzicht te verkrijgen in de mogelijkheden en beperkingen van PDE.



Figuur 6. Architectuur van een uitbreiding PDE voor expliciete foutmodellen

Referenties

- R. Davis en W. Hamscher**, 'Model-based troubleshooting', in *Exploring Artificial Intelligence*, Ed.: H.E. Shrobe, Morgan Kaufman Publishers, San Mateo, California, 1988, pp. 297-346. Ook in (Hamscher et al., 1992).
- W. Hamscher, L. Console en J. de Kleer (eds.)**, *Readings in model-based diagnosis*, Morgan Kaufman, San Mateo, CA, 1992
- J. de Kleer en B. Williams**, 'Diagnosing multiple faults', *Artificial Intelligence*, Vol. 32, 1987, pp. 97-130. Ook in (Hamscher et al., 1992).
- J. de Kleer en B. Williams**, 'Diagnosis with behavioral modes', in *Proceedings 11th IJCAI*, Detroit, 1989, pp. 1324-1330. Ook in (Hamscher et al., 1992).
- J. de Kleer**, 'Focusing on probable diagnoses', in *Proceedings 9th AAI*, Anaheim, 1991, pp. 842-848. Ook in (Hamscher et al., 1992).
- J. de Kleer, A. Mackworth en R.Reiter**, 'Characterizing diagnoses and systems', *Artificial Intelligence*, Vol. 56, 1992. Ook in (Hamscher et al., 1992).
- P. Struss en O. Dressler**, 'Physical negation: Integrating fault models into the general diagnostic engine', in *Proceedings 11th IJCAI*, Detroit 1989, pp. 1318-1323. Ook in (Hamscher et al., 1992).
- M.K. de Weger**, *Model-based diagnosis of dynamic systems with state*, Afstudeerverslag, Universiteit Twente, Enschede, 1992.