

Stysteemstoringen en menselijk gedrag

Citation for published version (APA):

Schaaf, van der, T. W. (1989). Stysteemstoringen en menselijk gedrag. In *Handboek Onderhoudsmanagement* (blz. I2020-1). (Handboek Onderhoudsmanagement, 1989).

Document status and date:

Gepubliceerd: 01/01/1989

Document Version:

Uitgevers PDF, ook bekend als Version of Record

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

Stroomstoringen en menselijk gedrag

T.W. van der Schaaf

Reprint Bdk/301

This article appeared in
Handboek Onderhoudsmanagement, Samsom,
Alphen a/d Rijn, oktober 1989, 1 -20.
Reprinted for private circulation by the author.



Eindhoven University of Technology
Department of Industrial Engineering and
Management Science

Postbox 513
5600 MB Eindhoven, Netherlands
Telephone 31-40-479111

Systeemstoringen en menselijk gedrag

Drs. T. W. van der Schaaf

1.	Inleiding	I2020- 2
1.1.	De menselijke factor bij bedrijfszekerheid	I2020- 2
1.2.	Het relatieve belang van de mens als faaloorzaak	I2020- 2
1.3.	Ingebakken bedieningsfouten	I2020- 3
2.	Menselijke fouten en automatisering	I2020- 4
2.1.	Paradox van de procesautomatisering	I2020- 4
2.2.	Processimulatie	I2020- 5
3.	Soorten menselijke fouten	I2020- 5
3.1.	Classificatie van Rasmussen	I2020- 5
3.2.	Een voorbeeld uit het verkeer	I2020- 6
3.3.	Gegevens uit de procesindustrie	I2020- 6
3.4.	Mogelijkheden voor preventie	I2020- 8
3.5.	Gebrek aan feedback	I2020- 9
4.	De twee kanten van de menselijke systeemcomponent	I2020- 9
4.1.	Human recovery	I2020- 9
4.2.	Preventie versus beheersing van fouten	I2020-10
4.3.	Ontwerpfactoren en human recovery	I2020-10
5.	Registratie- en analysesystemen in de industrie	I2020-11
5.1.	Ijsbergtheorie	I2020-11
5.2.	Problemen met de huidige systemen	I2020-12
6.	Beslissingsondersteuning bij storingsdiagnose	I2020-12
6.1.	Strategie mismatch	I2020-13
7.	Conclusies en aanbevelingen	I2020-14
	Literatuur	I2020-14
	Bijlage	I2020-16

Samenvatting

De bedrijfszekerheid van complexe systemen is niet alleen een zaak van *technisch* onderhoud, maar bevat ook belangrijke *menselijke* gedragscomponenten. Door af en toe te *falen* draagt deze menselijke factor bij aan een verhoogde kans op systeemstoringen. Daartegenover staat echter het unieke vermogen van mensen om van fouten te

leren. Bovendien kunnen mensen vaak onverwacht *herstellend* optreden, ook bij complexe en unieke systeemstoringen.

Deze aspecten worden besproken aan de hand van recente onderzoeksgegevens uit de Nederlandse procesindustrie. De consequenties hiervan voor het ontwerpen van systemen voor registratie en analyse van „menselijk falen”, bijna-ongelukken en menselijke herstelacties worden toegelicht.

In een bijlage tenslotte wordt aandacht besteed aan de mogelijkheden en problemen van geavanceerde technieken op dit gebied, zoals beslissingsondersteuning voor operators tijdens storingsdiagnose.

1. Inleiding

1.1. De menselijke factor bij bedrijfszekerheid

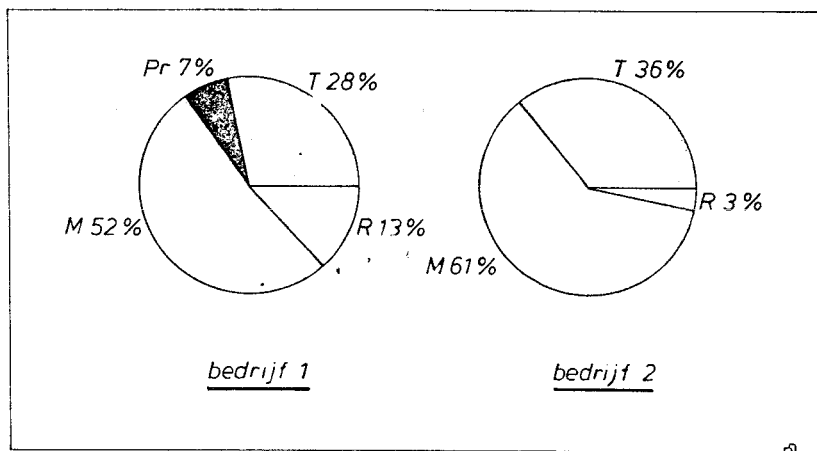
Productiebedrijven kunnen worden beschouwd als complexe mens-machine systemen. Dat wil zeggen dat de verschillende uit te voeren taken worden „verdeeld” tussen de component *mens* en de component *machine*, en wel liefst zodanig dat de sterke punten van elke component goed worden benut en de zwakke kanten niet of nauwelijks hoeven te worden aangesproken.

Tevens betekent dit dat er in principe drie mogelijke faaloorzaken zijn:

- de machine, bijvoorbeeld het ontwerp, de constructie of het onderhoud;
- de mens, bijvoorbeeld algemene menselijke beperkingen, slechte selectie, opleiding en training of lage motivatie;
- de interactie tussen mens en machine, bijvoorbeeld het interface tussen beide systeemcomponenten, maar ook de bedieningsvoorschriften en procedures.

1.2. Het relatieve belang van de mens als faaloorzaak

Een logische eerste vraag hierbij is of die menselijke invloed wel groot genoeg is om serieus te worden genomen bij het voorkomen van produktiestoringen (waarvan echte stops de meest ernstige vorm zijn). Uit figuur 1 blijkt dat het antwoord hierop ronduit „ja” moet zijn: in beide bedrijven (uit de Nederlandse procesindustrie) blijkt *menselijk falen* bij minstens de helft van de onderzochte storingen de hoofdoorzaak te zijn.



Figuur 1. Hoofdoorzaken van systeemfalen in de procesindustrie.

Op basis van 450 voorvallen in bedrijf 1 en 100 voorvallen in bedrijf 2
 M = falen van de mens Pr = falen van de procedures
 T = falen van de techniek R = restcategorie

Als tweede is „de techniek” als hoofdoorzaak van systeemfalen aan te wijzen: in dit geval wordt *technisch falen* als verzamelterm gebruikt voor alle fouten die in de ontwerpfase, constructiefase en het onderhoud zijn gemaakt.

Uit de „rest” categorie blijkt dat het met de registratie van al deze zaken in de industrie (te) vaak slecht is gesteld: ondanks veel extra naspeuringen onzerzijds bleek het regelmatig onmogelijk een zodanige reconstructie van het voorval te maken dat een classificatie in één van de drie soorten falen (menselijk, technisch of procedure falen) mogelijk was.

Procedurefalen tenslotte illustreert een oud probleem in de steeds veranderende productieomgeving: goede documentatie en aanpassing van bestaande procedures aan nieuwe apparatuur, enz. blijft bij sommige bedrijven een te lage prioriteit houden.

Conclusie: „menselijk falen” is een essentiële schakel bij het ontstaan van veel gevallen van systeemfalen.

1.3. Ingebakken bedieningsfouten

Wij moeten echter ook bedenken dat veel fouten eerder in de keten „ontwerp-constructie-onderhoud & bediening” pas naar boven komen in de *bedieningsfase* door operators. Oppervlakkig be-

schouwd leiden bijvoorbeeld vele ontwerpfouten (zoals van het mens-machine interface) tot ogenschijnlijk verkeerd storingsgedrag. De werkelijke hoofdoorzaak, zoals een ergonomisch onjuiste informatiepresentatie, komt dan pas uit de bus wanneer er, na herhaalde „bedieningsfouten”, structureel wordt onderzocht wat de *achterliggende factoren* bij die voorvallen waren. De latere bedieningsfouten zijn in de ontwerpfase vaak reeds „ingebakken”.

2. Menselijke fouten en automatisering

Het aloude gezegde „Vergissen is menselijk” moet zeer waarschijnlijk letterlijk worden genomen: *menselijke fouten vormen een natuurlijk onderdeel van menselijk gedrag als geheel*. Dat wil niet zeggen dat er nooit iets aan te doen valt, maar wel dat algemene waarschuwingen als „beter opletten” of „je mag geen fouten maken” weinig zoden aan de dijk zullen zetten.

Fouten maken moet!

Sterker nog: het is zelfs *zeer onwenselijk dat iemand nooit fouten mag maken*. Fouten maken levert immers feedback op met betrekking tot de grenzen van onze kennis en vaardigheden om een bepaald proces te bewaken, te besturen en te corrigeren.

In dit licht moeten de verhalen van operators worden gezien die met opzet (tijdens nachtdienst of in het weekend) kleine storingen in het proces genereerden om de effecten daarvan op andere variabelen uit te proberen. In de huidige sterk geautomatiseerde productiebedrijven waar operators nog „slechts” de supervisie hebben over de „software” regelkringen is dit experimenteren niet meer mogelijk: op het eerste gezicht misschien des te beter vanuit het oogpunt van proces-optimalisatie, maar op de lange termijn dodelijk voor de kennis en ervaring van operators om storingen op te lossen.

2.1. Paradox van de procesautomatisering

Bovendien worden veel kleine storingen zonder ingrijpen van de operator automatisch afgehandeld. Wanneer de automaat ooit mocht uitvallen of niet toereikend is voor een onverwachte, complexe storing kan de operator veelal niet meer zo snel en precies ingrijpen als vroeger. De *paradox van de procesautomatisering* luidt dan ook: Door hun zeldzaamheid worden storingen steeds minder hanteerbaar.

Kortom:

Automatisering die essentiële ervaring wegneemt bij de operator leidt mogelijk tot een *alles-of-niets situatie*: ofwel het automatisch gestuurde proces loopt perfect, beter dan men ooit met handbediening had kunnen bereiken; ofwel er treedt een storing op, die „bizar” en „complex” is (anders was er immers wel een software oplossing van tevoren bedacht) en die voor een topoperator al erg lastig is, maar voor iemand die nauwelijks storingsoplossende ervaring heeft natuurlijk volledig onhanteerbaar is, zodat een produktiestop dan onafwendbaar wordt.

2.2. Processimulatie

Een (gedeeltelijk) antwoord op bovenstaand probleem bestaat uit hoogwaardige faciliteiten voor *processimulatie*. Toch is dit zeker geen wondermiddel, aangezien men alleen datgene zal simuleren wat men voorstelbaar acht. Veel scenario's van werkelijk gebeurde „shut-downs” bevatten echter gedrags-elementen die men van tevoren nooit en te nimmer voor reëel zou hebben gehouden. Men maakt dus op zijn best *een gedeelte* van alle mogelijke storingssituaties mee, terwijl men zich deze beperking vaak niet realiseert. Hierdoor ontstaat een vals gevoel van veiligheid en beheersbaarheid, dat versluierend kan werken bij het oplossen van problemen, die net even anders zijn dan de trainingsopdrachten op de simulator.

3. Soorten menselijke fouten

Zoals hierboven al is gesteld, gaan we uit van menselijke fouten als natuurlijk onderdeel van menselijk gedrag. Gedrag op verschillende niveaus kan derhalve leiden tot verschillende soorten fouten. Eén van de meest invloedrijke modellen van operatorgedrag is dat van de Deense onderzoeker Jens Rasmussen.

3.1. Classificatie van Rasmussen

Zijn SRK (Skill-Rule-Knowledge based) model onderscheidt drie niveaus van gedrag (en dus ook fouten):

- het laagste niveau wordt *skill-based* genoemd en bestaat uit zeer goed geleerde automatische reacties op de omgeving zonder dat deze (veel) aandacht vereisen;
- het midden niveau heet *rule-based* en komt voor wanneer men weet hoe men moet handelen in een bepaalde situatie. Zo'n regel

kan in het geheugen zitten of als procedure „ter plekke” worden aangeboden en bevat een reeks stappen (die zich ieder op zich op skill-based niveau afspelen), gescheiden door checks op de correcte uitvoering ervan. Dit niveau van handelen vereist regelmatige of enige continue aandacht om foutloos gedrag te „garanderen”;

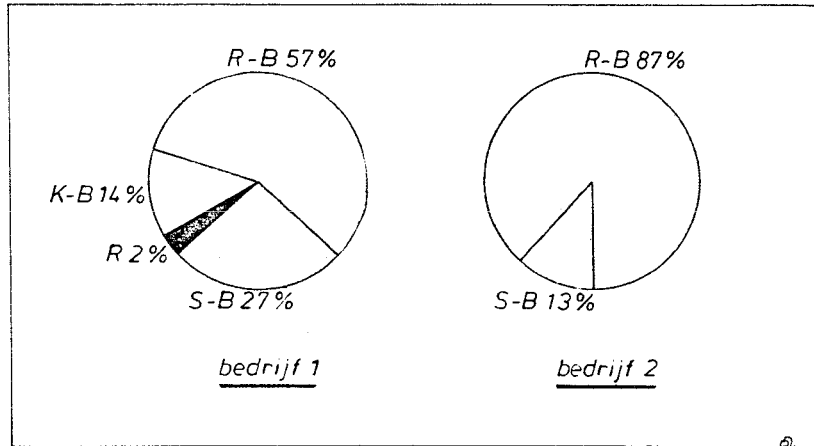
- tenslotte vinden we als hoogste niveau *knowledge-based* gedrag, dat wordt vertoond bij confrontatie met nieuwe, onbekende situaties waarvoor de persoon zelf oplossingen moet gaan vinden. Op dit niveau besteedt men alle beschikbare aandacht aan deze moeilijke taak.

3.2. Een voorbeeld uit het verkeer

Stelt u zich, om het bovenstaande model te verduidelijken, de volgende situaties voor: u nadert met de auto een kruispunt. Als er verkeerslichten staan en „uw” licht staat op rood dan zult u vrijwel automatisch en gedachteloos stoppen voor dat rode licht (skill-based gedrag). Op een gelijkwaardige kruising zonder stoplichten is uw gedrag wèl afhankelijk van het andere verkeer, omdat u nu even op moet letten om de hiervoor geldende regels (onder andere „rechts gaat voor”) correct toe te passen (rule-based gedrag). Als u tenslotte wordt geconfronteerd met een kruising waar het verkeer is vastgelopen omdat de verkeerslichten zijn uitgevallen, dan zult u zelf eerst een aantal beslissingen moeten nemen: bijvoorbeeld „ik heb haast en wil zo spoedig mogelijk verder”, of „ik rij in een geleende auto en wil in de eerste plaats blikschade voorkomen”. Daarna moet u iets verzinnen om dat doel te bereiken (omdraaien? zich ergens tussen wringen? wachten op verkeersagenten?) en dat plan uitvoeren. Een eventueel „moeilijk” gesprek met een passagier zal dan ongetwijfeld worden afgebroken omdat u alle aandacht nodig heeft om uit dit netelige probleem te komen (knowledge-based gedrag).

3.3. Gegevens uit de procesindustrie

De voorvallen uit figuur 1 waarbij menselijk falen de hoofdoorzaak was, zijn verder uitgesplitst naar de *aard* van die menselijke fouten. De resultaten van classificatie volgens het Rasmussen model zijn weergegeven in figuur 2.



Figuur 2. Aard van het menselijk falen in de procesindustrie.

Een verdere onderverdeling van de M-categorie in figuur 1 volgens het Rasmussen-model

S-B = skill-based falen

K-B = knowledge-based falen

R-B = rule-based falen

R = restcategorie

Net zoals in figuur 1 komen de resultaten van de twee bedrijven ruwweg overeen. Het meest frequent blijken rule-based fouten, waarbij vooral het bewust overtreden van veiligheidsregels opvalt. Daarna volgen de wat „simpeler” fouten op skill-based niveau, zoals struikelen, het niet opmerken van signalen, of het maken van typefouten en versprekingen in de communicatie met het proces c.q. collega's. Deze twee niveaus van fouten worden trouwens niet alleen relatief vaak geregistreerd omdat ze veel voorkomen, maar ook omdat ze vaak direct als fouten te herkennen zijn. Knowledge-based fouten zijn echter niet of pas na relatief lange tijd te detecteren en worden bovendien over het algemeen moeilijker gerapporteerd, waarschijnlijk omdat ze een grotere knauw betekenen in het zelfbeeld van kundige, ervaren procesoperators.

Juist omdat knowledge-based fouten lange tijd onopgemerkt kunnen blijven, zijn ze echter des te gevaarlijker. Wat betreft de kans op onverwachte produktiestops zou men zelfs kunnen stellen dat het vrij lage gevonden percentage van dit soort fouten een grotere aantasting van de bedrijfszekerheid vormt dan de veel frequentere skill- en rule-based fouten!

Een zeer belangrijke categorie binnen die knowledge-based fouten wordt gevormd door *verkeerde diagnoses* van storingsoorzaken.

Later in dit verhaal en in de bijlage zullen we daar nog uitgebreid op terugkomen.

3.4. Mogelijkheden voor preventie

Het praktische nut van dit soort classificaties moet natuurlijk liggen in de mogelijkheid om per type fouten de meest effectieve preventie toe te passen.

Preventie van *skill-based* fouten is vaak moeilijk vanwege het sterk automatische „openloop” karakter van gedrag op dit niveau: een typiste die bij iedere toetsaanslag zou gaan nadenken over de juistheid ervan wordt niet bepaald gewaardeerd op haar typesnelheid. Een ander soort fouten op dit niveau valt wel aan te pakken: al eerder hebben we immers gezien dat bepaalde (interface-)fouten al „ingebakken” worden in de ontwerpfase. Een *ergonomisch* verantwoord ontwerp van bijvoorbeeld beeldscherm informatie en toetsenborden kan een sterke reductie van het aantal fouten bewerkstelligen.

Rule-based fouten bestaan vaak uit verkeerde gewoonten („zonder beschermingsmiddelen werkt het sneller en er gebeurt mij toch nooit wat”). In zulke gevallen zou men kunnen denken aan het kunstmatig verminderen van de voordelen van zulk gedrag en tevens het benadrukken van de mogelijke negatieve consequenties. Ook kan men dan de voordelen en nadelen van het gewenste gedrag benadrukken c.q. verminderen.

Knowledge-based fouten kunnen worden veroorzaakt door gebrek aan (achtergrond-)kennis van inzicht in het te beheersen proces: (her-)training en selectie kunnen hier natuurlijk nuttig werk verrichten. Vaker is echter niet zozeer de kennis zelf de oorzaak, maar de manier waarop deze (verkeerd of slechts gedeeltelijk) wordt gebruikt bij het oplossen van problemen. Het denken in analogieën gaat bijvoorbeeld lang niet altijd op. Het slechts gedeeltelijk gebruiken van belangrijke informatie komt vaak voor in de vorm van de zogenaamde „bevestigingsneiging”: als men in het beginstadium van het probleemoplossen al een waarschijnlijke oplossing genereert, wordt vaak alle verdere informatie, die daarmee niet overeenstemt, genegeerd.

Selectief gebruik van informatie kan ook optreden wanneer de aandacht zich onder *stress* letterlijk vernauwt waardoor bepaalde informatie niet eens meer wordt waargenomen (zogenaamde *tunnelvisie*). Ook het werken in *groepen* kan soms eerder een blokkade betekenen voor alternatieve oplossingen (zogenaamde *grouptink*) dan een verbetering ten opzichte van een solistische oplossing.

3.5. Gebrek aan feedback

De gebrekkige feedback (laat en onduidelijk) naar aanleiding van knowledge-based fouten is waarschijnlijk de belangrijkste reden waarom deze categorie zeer moeilijk valt aan te pakken. Een mogelijke oplossing zou de zogenaamde *elektronische cocon* kunnen zijn: dit is een soort expertsysteem dat continu de menselijke operator superviseert bij diens probleemoplossend gedrag (bijvoorbeeld het beredeneren van een storingsoorzaak) en een signaal geeft wanneer conclusies getrokken worden die in strijd zijn met het geheel van bekende procesgegevens. Eenvoudiger versies hiervan in de vorm van ondersteunende hulpfuncties worden in de bijlage besproken.

4. De twee kanten van de menselijke systeemcomponent

„De mens als zwakste schakel in de keten” is een veelgehoorde opmerking. In het voorafgaande is inderdaad alleen de negatieve kant van de menselijke component in het totale systeem belicht: welk soort fouten maken mensen en hoe kunnen we die voorkomen?

We moeten er echter vanuit gaan, dat (bepaalde typen) fouten zich af en toe blijven voordoen, omdat ze tot het natuurlijke gedragsrepertoire behoren (zoals bij leerprocessen) en omdat preventie vaak erg moeilijk is (zoals bij diagnosefouten).

4.1. Human recovery

Het is dan ook belangrijk om tevens de positieve kant van afwijkend menselijk gedrag te bespreken: het (unieke) vermogen van mensen om originele oplossingen te vinden voor onverwachte problemen ten gevolge van fouten van henzelf of van collega's of veroorzaakt door falen van technische aard. Dit menselijk herstelvermogen (bestaande uit tijdige detectie van abnormale symptomen, juiste diagnose van de oorzaak en tenslotte een goed en tijdig uitvoeren van correctieve acties) wordt ook wel *human recovery* genoemd.

In de toekomst zal deze positieve kant van de mens alleen maar nog belangrijker worden, omdat de operatortaak steeds verder verschuift van regelmatig, via supervisor naar „trouble shooter” (zie: 2.1. Paradox van de procesautomatisering). De belangrijkste reden om de mens dan nog te handhaven als onderdeel van hoog-geautomatiseerde systemen bestaat juist uit diens creativiteit en flexibiliteit om onverwachte, nieuwe problemen te onderkennen en op te lossen

voordat automatische shut-down-mechanismen in werking treden of er anderszins schade of letsel ontstaat.

Dit betrekkelijk nieuwe inzicht wordt ondersteund door de resultaten van risico-analyses: het al of niet negeren van human recovery in foutenbomen bijvoorbeeld heeft een grotere invloed op de uiteindelijke systeemfaalkans dan alle objectieve faalgegevens van de technische componenten samen!

4.2. Preventie versus beheersing van fouten

Zeker wanneer preventie van fouten moeilijk is, terwijl human recovery zo'n grote potentiële invloed op de uiteindelijke systeemfaalkans heeft, moet aan deze laatste, nog onderbelichte, factor veel meer aandacht worden besteed. Een bepaald budget voor het bewaken en bevorderen van bedrijfszekerheid op systeemniveau zou zeker niet volledig aan preventie van storingen op componentniveau moeten worden besteed. Naast *preventie* wordt ook het *beheersbaar maken* van menselijke fouten en (kleine) technische storingen een effectief middel om de systeemfaalkans te verkleinen.

De achterliggende gedachte is dat het efficiënter kan zijn om bepaalde *fouten en storingen incidenteel te tolereren*, mits er in het systeemontwerp voor gezorgd is, dat deze *eventuele voorvallen beheersbaar* zijn, dat wil zeggen dat de operator in staat moet zijn om ze tijdig te detecteren, te diagnosticeren en te corrigeren.

Naast verder onderzoek naar het uitschakelen van faalfactoren moeten dus tevens de recoveryfactoren geanalyseerd en versterkt worden.

4.3. Ontwerpfactoren en human recovery

Rasmussen noemt zeer in het algemeen *observeerbaarheid* en *omkeerbaarheid* als voorwaarden voor human recovery. In de volgende *classificatie van ontwerpfactoren* wordt getracht dit verder uit te werken voor de procesindustrie:

- *technische ontwerpfactoren* kunnen de omkeerbaarheid (= de mogelijkheid en de tijd om nog correctief te kunnen optreden) optimaliseren. *Structurele* aspecten hierbij zijn bijvoorbeeld het gebruik van buffers tussen opeenvolgende procesonderdelen en het bestaan van alternatieve routes voor de produktstromen („lokale oplossingen voor lokale problemen”). *Dynamische* aspecten zijn vaak inherent aan het proces zelf en in dat geval vrijwel onveranderlijk. Het is daarom essentieel deze eventuele beperkingen met betrekking tot human recovery als uitgangs-

- punt te nemen en de rest van het ontwerp hieraan aan te passen: een vliegtuig komt nu eenmaal sneller in de problemen bij het uitvallen van de voortstuwing dan een schip;
- *interface ontwerpfactoren* bepalen vooral de observeerbaarheid van fouten en storingen. Bestrijding van alarminflatie bijvoorbeeld verhoogt de „transparantheid” van het detectiesysteem, terwijl redundantie van belangrijke meters de betrouwbaarheid van de gegevens op het interface bevordert;
 - *cognitieve ontwerpfactoren* tenslotte kunnen bestaan uit het implementeren van beslissingsondersteuning bij storingsdiagnosetaken (zie bijlage). Ook processimulatie kan worden gebruikt om operators op kunstmatige wijze ervaring op te laten doen met (zeldzame) storingsituaties en zodoende het inzicht in de procesdynamica te verhogen.

5. Registratie- en analysesystemen in de industrie

Een allereerste vereiste om menselijk falen en menselijk herstel in het eigen bedrijf aan te pakken is natuurlijk een *gedetailleerd inzicht* in de factoren die daar binnen de eigen poorten aan bijdragen. Voor (verzuim)ongevallen en incidenten met grote (productie)schade zijn over het algemeen wel systemen aanwezig waarmee een gedetailleerde beschrijving van het verloop en de gevolgen verzekerd is. De analyse daarna is vaak zeer specifiek en leidt dan ook tot typische ad-hoc maatregelen: een identiek voorval in de toekomst wordt dus vermeden.

5.1. Ijsbergtheorie

Paradoxaal genoeg bevinden bedrijven met relatief weinig ongevallen of onverwachte stops zich in een lastige positie: de feedback uit het eigen systeem is dan zeer mager, zodat men weinig gegevens krijgt om de situatie nog verder te optimaliseren. Hun „topje van de ijsberg” is met andere woorden erg klein. Onder de oppervlakte echter gebeuren talloze zaken (menselijke fouten en recoveries, kleine technische storingen) die niet tot enig blijvend resultaat leiden, maar die zich in wezen niet onderscheiden van de elementen die incidenteel wèl tot een ongeval of stop leiden. Het is dan dus zaak om ook die zogenaamde „*near-misses*” te laten rapporteren en analyseren. Bovendien heeft human recovery juist tot gevolg dat er geen of weinig zichtbare effecten zijn, waardoor deze belangrijke kant van de menselijke component vaak totaal onbelicht blijft, tenzij men serieus *near-misses* verzamelt.

5.2. Problemen met de huidige systemen

Vrijwel nooit heeft men een *verklarend model* of theorie van operatorgedrag om te bepalen welke informatie zou moeten worden geregistreerd en hoe men de resultaten daarna kan interpreteren. Hierdoor verzamelt men vaak veel overbodige gegevens zonder dat alle relevante informatie beschikbaar komt om effectieve maatregelen te nemen.

Een ander (praktisch) probleem is de *acceptatie door de werknemers* van dit soort zelfrapportagesystemen: het bedrijf vraagt alleen naar fouten en niet naar recoveries; er is soms een kans op sancties ten aanzien van de rapporteur; feedback over de behandeling van het rapport en over eventuele maatregelen bereikt de indiener vaak slechts gedeeltelijk, laat of indirect; tenslotte leidt een ad-hoc beleid vaak tot steeds weer nieuwe procedures in plaats van tot (duurdere) structurele oplossingen (zie 3.4. Skill-based fouten).

Het zal duidelijk zijn dat bovenstaande punten vermeden dienen te worden als men de belangrijkste informatiebron voor near-misses (de operators) mee wil laten werken. In een bedrijf waar *kwaliteitsdenken* door alle lagen van de organisatie ingevoerd, bevordert en geaccepteerd wordt, moet een nearmiss-rapportagesysteem als *leermiddel* echter zeker mogelijk zijn.

Eind juni 1989 werd rond deze nieuwe problematiek een EEG-workshop aan de Technische Universiteit Eindhoven georganiseerd waarvan de resultaten beschikbaar zullen komen voor het bedrijfsleven.

6. Beslissingsondersteuning bij storingsdiagnose

Al eerder hebben wij de mogelijk ernstige gevolgen van diagnosefouten benadrukt en gesteld dat dit probleem in de toekomst nog zal toenemen met de voortschrijdende procesautomatisering. Het lijkt dus wenselijk hierop te anticiperen door *adequate beslissingsondersteuning* voor operators te ontwikkelen. Deze zal af moeten hangen voor de *concrete informatiebehoefte* van de operator, die weer wordt bepaald door de *zoekprocessen* die het resultaat zijn van de keuze voor een bepaalde *diagnosestrategie*.

6.1. Strategie mismatch

Over de drie laatstgenoemde punten is op dit moment nog echter vrijwel niets bekend, waardoor het ontwikkelen van diagnose-ondersteuning een gevaarlijke zaak wordt. Indien immers de onderliggende strategie van zo een softwaresysteem niet overeenkomt met de jaren ervaring ingesleten (onbewuste) operators strategie(ën), dan kan zo'n mismatch de volgende consequenties hebben: ofwel de operators verwerpen het systeem als „niet-relevant” en gebruiken het niet, ofwel ze proberen hun eigen gedachtengang geforceerd aan te passen aan de software en vormen dan op zijn best een inefficiënt koppel.

Soortgelijke problemen hebben wij eerder in de procesindustrie geconstateerd bij de vele onzorgvuldige invoeringen van beeldschermen in bestaande paneelinstrumentatie-meet- en regelkamers. Bepaling van door operators gebruikte diagnosestrategieën is dus essentieel voor een adequate diagnose-ondersteuning.

Door middel van een aantal aan de Technische Universiteit Eindhoven ontwikkelde *abstracte diagnosetaken* is inmiddels enig inzicht verkregen in het scala van verschillende strategieën, de omstandigheden waaronder de ene strategie beter voldoet dan een andere, en in de vraag of operators makkelijk van strategie kunnen veranderen. In de bijlage wordt als voorbeeld één van die diagnosetaken uitgebreid behandeld.

Het voordeel van een inhoudsloze, algemene diagnosetaak is natuurlijk dat deze algemeen toepasbaar is en zinvol kan worden uitgevoerd door in principe elke operator met storingservaring; zodoende kunnen snel voldoende experimentele gegevens van praktijkmensen worden verzameld om betrouwbare conclusies te trekken. Het nadeel is echter dat de overeenkomst met echte diagnoses van storingen in het eigen proces niet bij voorbaat is gegarandeerd. Daarnaast zullen dus ook op andere manieren strategieën van operators geregistreerd en geëvalueerd moeten worden.

Daarom lijkt dit, in de toekomst steeds belangrijker vraagstuk bij uitstek geschikt om door industrie en universiteit samen te worden aangepakt: het theoretisch kader kan tenslotte pas echt worden getest in de praktijk en de praktijk kan deze nieuwe fundamentele inzichten vervolgens toepassen voor het verhogen van de bedrijfszekerheid.

7. Conclusies en aanbevelingen

- Menselijke bedieningsfouten spelen in de meeste gevallen van systeemfalen een (hoofd-)rol; een (onderschat) gedeelte daarvan wordt echter direct in de hand gewerkt door fouten in de ontwerpfase.
- Het is zinvoller om fouten te beheersen dan ze te verbieden. Beheersbaarheid van fouten is essentieel voor het optimaliseren van complexe mens-machinesystemen.
- Processimulatie biedt reële, maar beperkte mogelijkheden om de systeemkennisafbraak ten gevolge van automatisering tegen te gaan.
- De huidige registratiesystemen voor bedrijfsstoringen bieden onvoldoende inzicht in de aard en frequentie van menselijk falen, vooral wat betreft de meest ernstige vormen daarvan.
- Flexibiliteit en creativiteit maken de mens tot de zwakste en de sterkste schakel in de systeemketen.
- Bevorderen van „human recovery” moet, zeker in de toekomst, een even hoge (budgettaire) prioriteit krijgen als preventie van „human error”. Dit is alleen mogelijk door reeds in de ontwerp-fase een structurele inbreng van cognitief-ergonomische kennis te garanderen.
- In de procesindustrie verzamelt men te veel (irrelevante) en te weinig (gedrags)gegevens naar aanleiding van storingen en voorvallen, omdat een fundamentele basis voor een geïntegreerd registratie- en analysesysteem ontbreekt.
- „Near-miss” rapportage is onverenigbaar met zelfs de geringste vorm van sancties ten aanzien van de betrokkenen.
- Het invoeren van diagnose-ondersteunende systemen zonder deze af te stemmen op de operatorstrategieën, is op zijn minst inefficiënt en waarschijnlijk zelfs risico verhogend.

Literatuur

Algemene principes van menselijk gedrag:
Leyden, J. sr., van, „*Psychologische functieleer*”, 2e druk 1989; Van Loghum Slaterus, Deventer.

„Human error”, „human recovery” en risico analyse:
Goossens, L. H. J. (ed.), „*Human recovery*”: Proceedings of the COST A 1 Seminar, oktober 1988, Technische Universiteit Delft, Vakgroep Veiligheidskunde.

Operatortaken in de procesindustrie:

Kragt, H. en Moraal, J., „*Meet- en regelkamers in de procesindustrie: operators waardig?*”, 1984, PT Procestechniek, no. 39, p. 1-4.

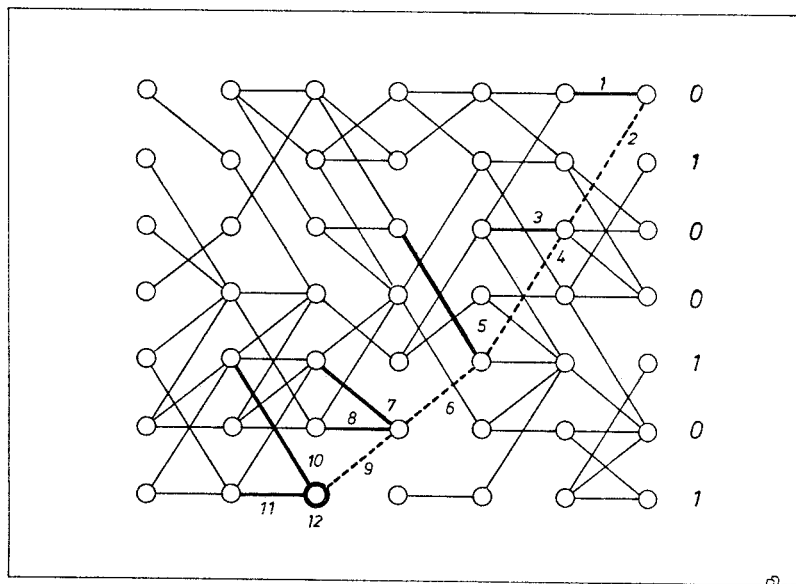
Moraal, J., „*Dealing with complexity; the process operator in the industrial setting*”. November 1988, rapport Technische Universiteit Eindhoven, vakgroep Technologie en Arbeid, sectie Ergonomie, nr. 37.

BIJLAGE

Beslissingsondersteuning bij een abstracte storingslokalisatietaak

Drs. T. W. van der Schaaf, Technische Universiteit Eindhoven

Door de vakgroep Technologie en Arbeid, sectie Ergonomie is een aantal abstracte diagnosetaken ontwikkeld. Deze worden gebruikt voor het onderzoek naar de strategieën die operators toepassen voor storingslokalisatie.



Figuur a. Voorbeeld van een netwerk met daarin aangegeven een Tracing-Back strategie voor storingslokalisatie.

De cijfers in het netwerk geven de volgorde aan waarin verbindingen zijn getest; een vette lijn betekent testuitslag = 1, een *stippellijn* betekent testuitslag = 0. De falende component (= test 12) is aangegeven door een vette cirkel.

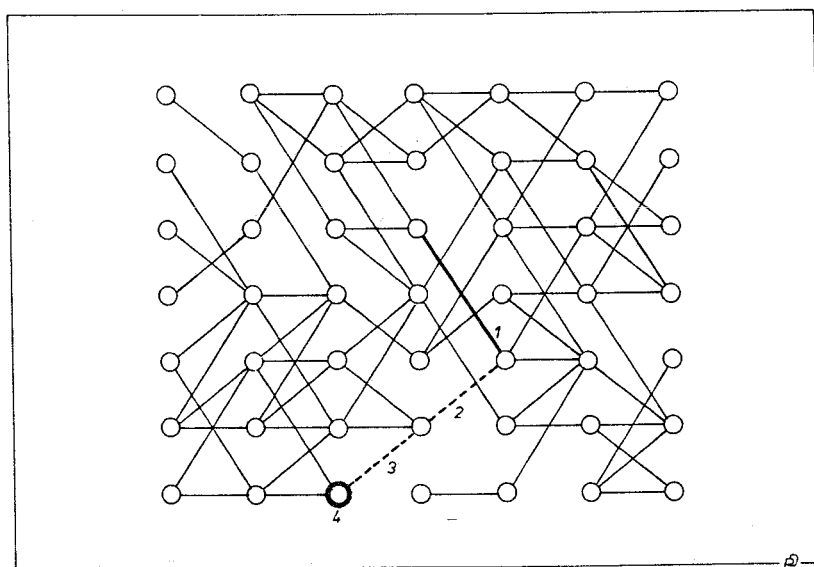
Netwerктаak

Als voorbeeld wordt in figuur a. één van deze taken als een netwerk van componenten (rondjes) met hun onderlinge verbindingen (lijntjes) getoond. In dit statische systeem planten (elektrische) signalen zich voort van links naar rechts, waarbij de componenten

als „en”-poorten fungeren. Dit betekent dat een intacte component alleen dan een „outputsignaal” (= 1) geeft, indien *alle* inputs ook de waarde 1 hebben. In elk netwerk is steeds precies één component defect waardoor deze falende component „geen signaal” (= 0) als outputs geeft. Aangezien dit één of meerdere nulinputs veroorzaakt voor verder naar rechts gelegen componenten ontstaat er een patroon van nul- en één-outputs en -inputs. Tenslotte resulteert dit in de metingen zoals die voor de meest rechtse kolom van componenten worden gegeven aan het begin van de taak. De proefpersoon moet de ene falende component vinden en kan daarvoor (tegen bepaalde kosten) informatie over de status (1 of 0) van verbindingen en componenten uit het netwerk opvragen.

Twee strategieën

Uit experimenten met studenten en procesoperators als proefpersoon blijkt dat zelfs in deze relatief simpele netwerken zéér uiteenlopende strategieën worden gebruikt. Twee uitersten worden getoond in figuur a en b.



Figuur b. Idem figuur a, echter nu met de hypothese en teststrategie aangegeven.

Figuur a. toont een makkelijk uit te voeren strategie waarbij de proefpersoon een willekeurig „spoor” van aaneengesloten „nul”-verbindingen zoekt om zo uiteindelijk, na veel testinformatie te hebben opgevraagd, uit te komen bij de falende component. Dit wordt de *tracing back* (TB) strategie genoemd.

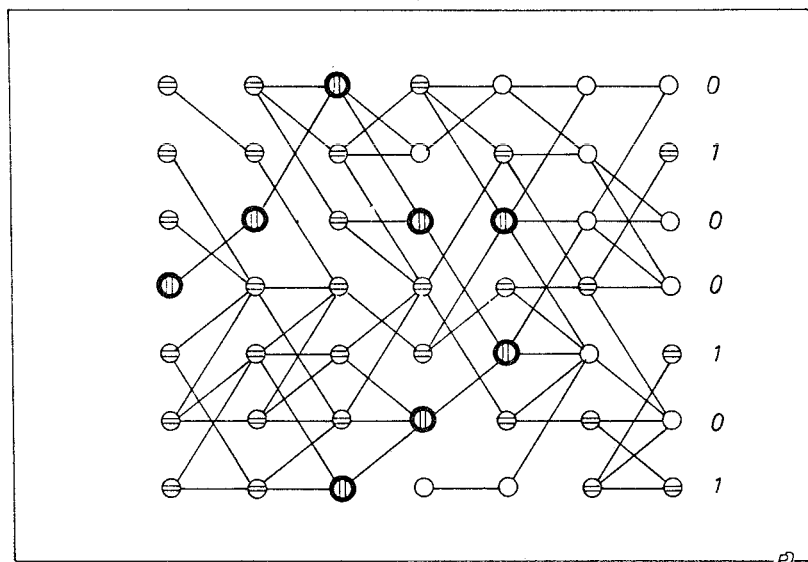
In figuur b. wordt daarentegen juist uiterst spaarzaam getest. Deze efficiency is echter slechts te bereiken ten koste van zeer grote *mentale belasting*, want bij iedere stap moet *alle* informatie die dan bekend is opnieuw worden bekeken en onthouden. Daarna volgt een test waarmee de verzameling van mogelijk falende componenten optimaal wordt gereduceerd. De testuitslag (1 of 0) leidt dan weer tot een algehele heranalyse van deze nieuwe situatie, waarna weer een optimale test volgt, enz.

Deze zogenaamde *hypothese en test* (HT) strategie kenmerkt zich dus door lange denktintervallen tussen een beperkt aantal tests, terwijl TB bestaat uit een lange reeks tests die elkaar snel opvolgen. Een ander belangrijk verschil is de hoeveelheid informatie die per test beschikbaar moet zijn: HT vereist steeds de beschikbaarheid van het *gehele* netwerk met *alle* tot dan toe verzamelde informatie, terwijl voor TB slechts zeer beperkte, lokale informatie nodig is: de component waar men aanbeland is, plus alleen de *directe* inputverbindingen daarvan.

Beslissingsondersteuning

Het zal duidelijk zijn dat deze strategieën zeer uiteenlopende informatiebehoefte impliceren en dus ook verschillende vormen van beslissingsondersteuning. Voor TB is bijvoorbeeld een simpele hulpfunctie die alle directe inputs naar bepaalde componenten aangeeft al voldoende. Het achtereenvolgens optimaal reduceren van de verzameling van mogelijk falende componenten totdat deze verzameling nog slechts één component telt, vereist echter een combinatie van op zich al complexere hulpfuncties. Zo'n HT wordt bijvoorbeeld pas echt ondersteund door uit twee subsets van het netwerk samen de bovengenoemde verzameling zichtbaar te maken (zie figuur c.).

Subset 1 (= *verticale* arcering) wordt berekend door een hulpfunctie die nagaat welke componenten met *alle* nultuitgangen in de meest rechtse kolom zijn verbonden. Subset 2 (= *horizontale* arcering) is het resultaat van een hulpfunctie die alle componenten aangeeft die met minstens een éénuitgang in de meest rechtse kolom zijn verbonden. Subset 1 minus subset 2 geeft de gevraagde verzameling, extra aangeduid met vette cirkels.



Figuur c. Voorbeeld van beslissingsondersteuning bij het bepalen van de verzameling mogelijk falende componenten aan het begin van de taak.

De subset van componenten die met *alle* „nultgangen” geheel rechts in het netwerk zijn verbonden, minus de subset van componenten die minstens met een éénuitgang” geheel rechts zijn verbonden, definieert de gewenste verzameling van mogelijk falende componenten. De mate waarin een door de proefpersoon gevraagde test die verzameling reduceert, is een duidelijke indicatie voor de juistheid van de keuze voor die test.

De optimale strategie bestaat niet

Het zou een groot misverstand zijn te denken dat hypothese en teststrategie (HT) altijd een „betere” strategie is dan tracing back strategie (TB). In de praktijk zou het tegendeel eerder waar kunnen zijn! Een belangrijke reden hiervoor is de factor *tijdsdruk*. Vaak blijkt TB sneller tot een oplossing te leiden dan HT: de grotere hoeveelheid uit te voeren tests op het systeem bij TB wordt in dat geval meer dan gecompenseerd door de snelheid waarmee die tests elkaar opvolgen. Wanneer informatie inwinnen echter moeilijk is, ofwel gepaard gaat met zeer hoge kosten, dan is HT natuurlijk weer aantrekkelijk.

I2020-20 Systeemstoringen en menselijk gedrag

Kortom: het proces, de aard van de storingsituatie én de operators bepalen samen wat „de” optimale diagnose strategie is en dus ook hoe de operators bij deze taak moeten worden ondersteund.