

H4304

Opdrachtgever:

DG Rijkswaterstaat

Rijksinstituut voor Kust en Zee

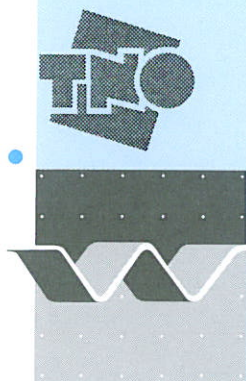
## Haalbaarheidsstudie HYDRA-K

Fase 2 – Inbouwen van additionele faalmechanismen en bekledingstypen in  
het demonstratiemodel

H.R.A. Jagers, F.L.M. Diermanse, M. Klein-Breteler, A.C.W.M.  
Vrouwenvelder, H.M.G.M. Steenbergen

rapport

november 2003



Opdrachtgever:

DG Rijkswaterstaat

Rijksinstituut voor Kust en Zee

## Haalbaarheidsstudie HYDRA-K

Fase 2 – Inbouwen van additionele faalmechanismen en bekledingstypen in  
het demonstratiemodel

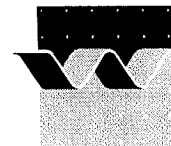
H.R.A. Jagers, F.L.M. Diermanse, M. Klein-Breteler, A.C.W.M.  
Vrouwenvelder, H.M.G.M. Steenbergen

rapport

november 2003



**wl | delft hydraulics**



OPDRACHTGEVER: DG Rijkswaterstaat, RIKZ

TITEL: Haalbaarheidsstudie HYDRA-K  
Fase 2 – Inbouwen van additionele faalmechanismen en bekledingstypen in het demonstratiemodel

SAMENVATTING:

In het kader van de voorbereidingen voor het hydraulische randvoorwaardenboek wordt momenteel een rekenprogramma ontwikkeld door het Rijksinstituut voor Kust en Zee (RIKZ). Door middel van het inbouwen van dll's zal HYDRA-K op termijn uitgebreid worden met een aantal faalmechanismen, behorende bij de volgende bekledingstypen steenzetting, asfaltbekleding en grasbekleding. Echter, tijdens de uitvoering van deze studie zijn de dll's nog niet beschikbaar. Daarom worden de rekenregels waarop de dll's gebaseerd zijn ingeprogrammeerd in een speciaal te bouwen demonstratiemodel.

Het doel van dit project is om de haalbaarheid van de inbouw van de genoemde faalmechanismen/bekledingstypen in HYDRA-K te onderzoeken. Dit rapport beschrijft fase 2 van het project, waar de faalmechanismen daadwerkelijk zijn ingebouwd in het demonstratiemodel. Na voltooiing van de implementatie zijn testberekeningen uitgevoerd voor de nieuwe faalmechanismen en bekledingstypen met het model. Voor drie locaties (Vlissingen, Den Helder en Petten) zijn in eerste instantie ontwerpberekeningen uitgevoerd voor dijkprofielen met uniforme bekledingstypen. Geen van de resultaten bleek in strijd met datgene wat op voorhand verwacht kon worden. Het model lijkt derhalve betrouwbare uitkomsten te leveren.

Vervolgens zijn ontwerpberekeningen uitgevoerd voor delen van de dijk en toen bleek dat het model in een aantal situaties geen uitkomst geeft. Met name voor laaggelegen delen wordt het fenomeen van de zogenaamde "niet-reguliere faaldomeinen" waargenomen. De hydraulische belasting is in dit geval niet meer een strikt stijgende functie van de waterstand en windrichting, zodat de simultane waarnemingen bij een toename in de grootte van de verschuiving uit het faaldomein kunnen verschuiven. In dit rapport zijn twee oplossingsrichtingen aangedragen om te realiseren dat ook voor niet-reguliere faaldomeinen faalfrequentie-berekeningen en ontwerpberekeningen uitgevoerd kunnen worden. Beide methoden worden verder uitgewerkt en geanalyseerd in de vervolgfases van dit project.

REFERENTIES:

VER.	AUTEUR	DATUM	OPMERK.	REVIEW	GOEDKEURING
01	B. Jagers et al.	20-11-2003		F. Diermanse	W. Tilmans

PROJECTNUMMER H4304

TREFWOORDEN: HYDRA-K, randvoorwaarden, belastingmodellen, faalkansen, kustsystemen, leidraad toetsen

AANTAL BLADZIJDEN 73

VERTROUWELIJK  JA, tot (datum)  NEE

STATUS:  VOORLOPIG  CONCEPT  DEFINITIEF

## Inhoud

<b>1</b>	<b>Inleiding .....</b>	<b>1—1</b>
<b>2</b>	<b>Nieuwe faalmechanismen en bekledingstypen in HYDRA-K .....</b>	<b>2—1</b>
2.1	Inleiding .....	2—1
2.2	Grenstoestandsfunctie voor asfaltbekleding .....	2—3
2.3	Grenstoestandsfunctie voor grasbekleding .....	2—5
2.4	Grenstoestandsfunctie voor steenzetting.....	2—8
<b>3</b>	<b>Implementatie van de nieuwe functionaliteiten.....</b>	<b>3—1</b>
3.1	Overwegingen en keuzes .....	3—1
3.1.1	Omgaan met meerdere typen bekledingen op één dijkprofiel ...	3—1
3.1.2	Omgaan met het stormverloop.....	3—1
3.1.3	Stormduur als additionele stochast .....	3—4
3.2	Beschrijving van de implementatie.....	3—6
3.2.1	Inleiding .....	3—6
3.2.2	Z-functies .....	3—7
3.2.3	Aanroep van Z-functies.....	3—8
3.2.4	Faalfrequentie-berekeningen.....	3—9
3.2.5	Ontwerpberekeningen .....	3—9
3.2.6	Algemene routines .....	3—10
3.2.7	User interface .....	3—10
3.2.8	Batch routines .....	3—11
3.2.9	Overige aanpassingen aan de code.....	3—11

<b>4</b>	<b>Beknopte gebruikershandleiding voor de nieuwe faalmechanismen .....</b>	<b>4—1</b>
4.1	Inleiding.....	4—1
4.2	User interface.....	4—1
4.3	Batchberekeningen .....	4—3
<b>5</b>	<b>Testberekeningen .....</b>	<b>5—1</b>
5.1	Inleiding.....	5—1
5.2	Uniforme bekleding, optie hoogste waterstand .....	5—1
5.3	Uniforme bekleding, optie ongunstigste belasting .....	5—5
5.4	Frequentieberekeningen .....	5—6
5.5	Berekeningen voor delen van het profiel.....	5—6
<b>6</b>	<b>Niet-reguliere faaldomeinen .....</b>	<b>6—1</b>
6.1	Probleembeschrijving .....	6—1
6.1.1	Illustratie aan de hand van een rekenvoorbeeld.....	6—1
6.1.2	Probleembeschrijving algemeen.....	6—4
6.2	Mogelijke oplossingsrichtingen.....	6—4
6.2.1	Andere verschuiving in het standaard exponentiële vlak .....	6—4
6.2.2	Kunstmatige verlenging van stormgebeurtenissen .....	6—6
<b>7</b>	<b>Conclusies.....</b>	<b>7—1</b>
<b>8</b>	<b>Referenties.....</b>	<b>8—1</b>
<b>Bijlage:</b>		
<b>A</b>	<b>Aanpassingen in de code .....</b>	<b>A—1</b>

# I Inleiding

## Achtergrondinformatie

In het kader van de voorbereidingen voor het hydraulische randvoorwaardenboek wordt momenteel een rekenprogramma ontwikkeld door het Rijksinstituut voor Kust en Zee (RIKZ). De voornaamste functionaliteiten van dit rekenprogramma (HYDRA-K) zijn:

- het berekenen van hydraulische randvoorwaarden langs de kust, die bestaan uit een combinatie van de waterstand en golfparameters (het zogenaamde “ontwerppunt” of “toetspunt”) en dienen als invoer voor een toets op de veiligheid;
- het berekenen van een faalkans van een waterkering; en
- het berekenen van de benodigde sterkte en/of dimensie van de waterkering voor een gegeven veiligheids criterium.

Genoemde functionaliteiten zijn momenteel binnen HYDRA-K ontwikkeld voor drie faalmechanismen:

- golfploop;
- golfoverslag; en
- stabiliteit (van steenzetting).

In het kader van de voorbereidingen op het hydraulische randvoorwaardenboek zal HYDRA-K uitgebreid worden met een aantal faalmechanismen, behorende bij de volgende bekledingstypen:

- steenzetting (een uitbreiding van de huidige functionaliteit);
- asfaltbekleding; en
- grasbekleding.

Op langere termijn worden de faalmechanismen door middel van dll's in HYDRA-K ingebouwd. Tijdens de uitvoering van deze studie zullen de dll's nog niet beschikbaar zijn, maar de rekenregels waarop ze gebaseerd zijn, staan beschreven in de leidraad toetsen op veiligheid (TAW, 1999).

## Probleemstelling

Het toevoegen van de in de inleiding genoemde additionele faalmechanismen en bekledingstypen aan HYDRA-K behelst meer dan alleen het inbouwen van een aantal dll's. De belastingelementen van de additionele faalmechanismen/bekledingen verschillen namelijk van de belastingelementen van de faalmechanismen/bekledingstypen die reeds zijn ingevoerd in HYDRA-K. Eerst dient daarom vastgesteld te worden of de additionele faalmechanismen/bekledingstypen binnen het bestaande concept van HYDRA-K passen.

Daarbij spelen de volgende vraagstukken:

- welke aanpassingen zijn nodig binnen HYDRA-K om de additionele faalmechanismen in te passen?
- hoeveel inspanning vergen deze aanpassingen?
- zijn alle gewenste aanpassingen mogelijk binnen het concept van HYDRA-K?
- is het mogelijk om middels work-arounds alternatieve oplossingen te vinden voor dié aanpassingen die niet binnen het concept van HYDRA-K passen?
- is het op langere termijn technisch haalbaar om, middels enige wijzigingen aan het concept van HYDRA-K, alsnog alle gewenste aanpassingen te realiseren?
- welke inspanningen zijn nodig om een derde (of eventueel vierde) stochast aan HYDRA-K toe te voegen?

### Doel van het project

Het doel van dit project is om de haalbaarheid van de inbouw van de genoemde faalmechanismen/bekledingstypen in HYDRA-K te onderzoeken. Afhankelijk van de uitkomst van de haalbaarheidsstudie zal een aantal van deze faalmechanismen in een demonstratiemodel van HYDRA-K ingebouwd worden. De nauwkeurigheid van de rekenresultaten van deze faalmechanismen zal geanalyseerd worden.

Voor de faalmechanismen die niet op korte termijn ingebouwd kunnen worden, wordt in de eerste plaats een work-around opgesteld en vervolgens wordt onderzocht of het op langere termijn technische haalbaar is om, middels enige wijzigingen aan het concept van HYDRA-K, deze faalmechanismen alsnog in te bouwen.

### Fasering

Het project is opgedeeld in vijf fasen:

Fase 1: Plan van aanpak.

**Fase 2: Inbouwen van additionele faalmechanismen en bekledingstypen in het demonstratiemodel.**

Fase 3: Indicatieve bepaling van de nauwkeurigheid.

Fase 4: Alternatieven (work-arounds) voor het inbouwen van faalmechanismen.

Fase 5: Mogelijkheden voor de inbouw van faalmechanismen in HYDRA-K op langere termijn en effecten op de hydraulische randvoorwaarden.

Het voorliggende rapport vormt één van de eindproducten van **fase 2**: “*Inbouwen van additionele faalmechanismen en bekledingstypen in het demonstratiemodel*”. De volgende activiteiten zijn uitgevoerd in fase 2:

1. inbouwen van de additionele bekledingstypen met bijbehorende faalmechanismen in een demonstratiemodel van HYDRA-K;
2. visualiseren van de grenstoestandsfunctie voor elk van de ingebouwde faalmechanismen en bekledingstypen; en
3. monitoren van problemen en aandragen van oplossingen.

## Leeswijzer

De lezer wordt verondersteld kennis te hebben van de achtergronden en toepassingsgebieden van HYDRA-K. Indien dat niet het geval is, wordt aangeraden eerst het functioneel ontwerp (HKV, 2000) te bestuderen.

Hoofdstuk 2 van dit rapport beschrijft de verschillende faalmechanismen en bekledingstypen die in het demonstratiemodel geïmplementeerd zijn. Hoofdstuk 3 beschrijft vervolgens welke aanpassingen in de code zijn gerealiseerd. In hoofdstuk 4 wordt een beknopte gebruikershandleiding gegeven van de nieuwe elementen van het demonstratiemodel. Hoofdstuk 5 beschrijft de testberekeningen die met het model zijn uitgevoerd. In hoofdstuk 6 wordt ingegaan op het fenomeen van de “niet-reguliere faaldomeinen” waar in sommige testberekeningen tegenaan gelopen is. Hoofdstuk 7, ten slotte, bevat de belangrijkste conclusies van dit rapport.

## Auteurs en organisatie

Dit rapport is opgesteld door dr. ir. H.R.A. Jagers (WL | Delft Hydraulics), dr. ir. F. L. M. Diermanse (WL | Delft Hydraulics, projectleider), ir. M. Klein Breteler (WL | Delft Hydraulics), Prof. ir. A.C.W.M. Vrouwenvelder (TNO-Bouw) en ir. H.M.G.M. Steenbergen (TNO-bouw).

Het voorliggende rapport is één van de producten van het project, zoals verwoord in RIKZ-opdrachtnummer RKZ-1345 d.d. 15 september 2003.



## 2 Nieuwe faalmechanismen en bekledingstypen in HYDRA-K

### 2.1 Inleiding

Op basis van de leidraad toetsten (TAW, 1999) komen de volgende faalmechanismen in aanmerking om aan het demonstratiemodel van HYDRA-K toegevoegd te worden:

- opdrukken van asfalt;
- bezwijken van steenzetting of asfalt op een berm;
- lokale afschuiving bij steenzetting;
- materiaal transport bij steenzetting;
- schade aan de bekleding als gevolg van golfklappen op het buitentalud (voor gras, asfalt en steenzetting); en
- schade aan de grasbekleding als gevolg van stroming op het buitentalud.

Tijdens fase 1 van dit project is besloten om drie van deze faalmechanismen *niet* aan het demonstratiemodel worden toegevoegd, en wel om de volgende redenen:

- *Materiaaltransport*: Dit faalmechanisme wordt niet meegenomen omdat het niet afhankelijk is van de golfhoogte, periode of waterstand. Bovendien is dit faalmechanisme meestal niet maatgevend.
- *Opdrukken van asfalt*: Voor dit faalmechanisme is een goed inzicht in de freatische lijn in de dijk nodig, hetgeen niet beschikbaar is.
- *Bezwijken van steenzetting of asfalt op een berm*: Faalmechanismen voor bermen wijken dusdanig af van faalmechanismen op het talud dat voorafgaand aan het project met de opdrachtgever is afgesproken om deze niet in het demonstratiemodel op te nemen. Asfalt op een berm wordt niet meegenomen omdat er geen rekenregels staan in de leidraad toetsen (TAW, 1999) en in het 'technisch rapport asfaltconstructies' van de TAW (VTV, 2003).

Voor de drie genoemde faalmechanismen wordt in fase 4 van dit project een work-around opgesteld. De overige faalmechanismen zijn wel ingevoerd.

In dit hoofdstuk wordt de theorie beschreven achter het belangrijkste onderdeel van de implementatie van faalmechanismen/bekledingstypen, nl. de grenstoestandsfuncties. Op de lange termijn worden deze grenstoestandsfuncties door middel van dll's in HYDRA-K ingebouwd. Tijdens de uitvoering van deze haalbaarheidsstudie zijn deze dll's nog niet beschikbaar en worden als alternatief de formules uit de leidraad toetsen (TAW, 1999) geprogrammeerd in het demonstratiemodel.

## De rol van grenstoestandsfuncties binnen HYDRA-K

De methode waarmee binnen HYDRA-K de faalkans van een kering wordt berekend, staat uitgebreid beschreven in het functionele ontwerp (HKV, 2000). Samengevat worden de volgende stappen uitgevoerd:

- stap 1.** Inlezen van simultane waarnemingen van wind en waterstand.
- stap 2.** Selectie van de hoogste waterstanden tijdens elke stormgebeurtenis met de bijbehorende windsnelheid (deze stap wordt alleen uitgevoerd als de gebruiker de optie “hoogste waterstand” geselecteerd heeft. Indien de optie “ongunstigste belasting” geselecteerd is wordt de gehele meetreeks meegenomen).

Vervolgens worden voor elk van de geselecteerde simultane waarnemingen de volgende stappen uitgevoerd:

- stap 3.** Opschalen van de windsnelheid en de waterstand.
- stap 4.** Bepalen van de hydraulische belasting voor de teen van de dijk (waterstand, golfhoogte, -periode en -richting) op basis van de opgeschaalde waarden van windsnelheid en waterstand.
- stap 5.** Vaststellen of de hydraulische belasting groter is dan de weerstand van de kering, in welk geval de kering verondersteld wordt te falen.

Na voltooiing van deze vijf stappen wordt geteld voor hoeveel opgeschaalde stormgebeurtenissen falen optreedt, op basis waarvan de jaarlijkse faalfrequentie van de kering wordt vastgesteld.

De grenstoestandsfuncties worden in stap 5 van de bovenstaande procedure aangeroepen. De generieke beschrijving van grenstoestandsfuncties van bekledingen is van de vorm:

$$Z = R(D, L, \dots) - S(\Delta h, H_s, T_s \text{ of } T_p)$$

Waarin:

Z	= grenstoestandsfunctie
R	= sterkte
S	= belasting
L	= typische afmeting bekledingsonderdeel
D	= dikte van de bekleding
...	= andere parameters
$\Delta h$	= $h - h_A$
h	= stilwaterstand (t.o.v. NAP)
$h_A$	= hoogte van het beschouwde onderdeel van de bekleding (t.o.v. NAP)
$H_s$	= significante golfhoogte
$T_s$ of $T_p$	= maat voor de golfperiode

Falen van de kering wordt geacht op te treden als Z kleiner is dan nul; bij positieve Z is de constructie nog in tact.

In de paragrafen 2.2 tot en met 2.4 wordt per bekledingstype een beschrijving gegeven van de grenstoestandsfuncties die aan het demonstratiemodel van HYDRA-K zijn toegevoegd.

## 2.2 Grenstoestandsfunctie voor asfaltbekleding

De methode voor het vaststellen van de betrouwbaarheid van asfaltbekledingen is gebaseerd op de leidraad Toetsen op Veiligheid (TAW, 1999). De stabiliteitsformules voor asfalt zijn alleen gegeven voor het belastingtype golfklappen. Het faalmechanisme kan het beste omschreven worden als “*schade aan de bekleding als gevolg van golfklappen*”.

Voor een correcte toetsing ten aanzien van het faalmechanisme “*opdrukken van asfalt*” zou een goed inzicht van de freatische lijn in de dijk nodig zijn. Daarom wordt er in eerste instantie van afgezien om dit mechanisme in het demonstratiemodel onder te brengen. In fase 4 van dit project zal wel getracht worden een work-around op te stellen.

De volgende subtypen asfaltbekledingen zijn beschouwd:

- waterbouwasfalt beton (Type 1);
- open steenasfalt (Type 5).

Daarnaast wordt onderscheid gemaakt in twee typen ondergrond:

- zand;
- klei.

Voor de toetsing van asfaltbekledingen zijn de volgende invoer grootheden nodig:

$H_s$	:	significante golfhogte op de teen van de constructie (m)
$D$	:	dikte van de asfaltbekleding (m)
$h$	:	waterstand (m + NAP)
$z_o$	:	niveau ondergrens van de te toetsen bekleding (m + NAP)
$z_b$	:	niveau bovengrens van de te toetsen bekleding (m + NAP)
$\tan \alpha$	:	taludhelling (-)
Type ondergrond	:	zand of klei
Type asfalt	:	waterbouwasfalt beton, Open steenasfalt
Type filter (bij open steenasfalt)	:	geotextiel, zandasfalt

De toetsing start met een beoordeling of het stuk bekleding wel belast wordt:

Bovengrens:	als $h < z_o - \frac{1}{4} H_s$	=> bekleding faalt niet ( $Z > 0$ ) (fig. 3.2.2.1 LTV'99)
Ondergrens:	als $h > z_b + 2H_s$	=> bekleding faalt niet ( $Z > 0$ )

Als de waterstand tussen deze bovengrens en ondergrens ligt wordt verder getoetst, nl. de laagdikte. De eisen ten aanzien van de laagdikte zijn afhankelijk van het type asfalt.

### Waterbouwasfaltbeton

De algemene formule voor het bepalen van de minimaal benodigde laagdikte wordt weergegeven met vergelijking (1):

$$D_{\min} = \max \{ D_a ; C_3 + C_4 H_s \} \quad (1)$$

De waarde van  $D_a$  is een functie van golfhoogte  $H_s$  en taludhelling  $\tan \alpha$ . De functiebeschrijving is afhankelijk van het type onderliggende laag (zand of klei) en gebaseerd op de leidraad toetsen (zie fig. 3.2.2.6 uit de leidraad):

[1] waterbouwasfaltbeton met ondergrond zand:

$$D_a = (-0.18401 + 0.125056 H_s + 0.0014 H_s^2) (-1.69947 + 5.174361 \tan \alpha + 0.098738 H_s) \quad (2)$$

[2] waterbouwasfaltbeton met ondergrond klei:

$$D_a = (-0.06927 + 0.084858 H_s + 0.013069 H_s^2) (-0.93087 + 5.108631 \tan \alpha + 0.060252 H_s) \quad (3)$$

Beide functiebeschrijvingen worden ook toegepast in het softwarepakket PC-Ring (TNO-bouw, 2003).

De waarden voor  $C_3$  en  $C_4$  zijn:

$$\begin{aligned} C_3 &= 0,1 \\ C_4 &= 0,0167 \end{aligned}$$

Deze toetsingsmethode is alleen geldig wanneer de golfhoogte  $H_s$  kleiner is dan 6,0 m (zie fig. 3.2.2.6 uit de leidraad toetsen). Dit betekent dat alleen een score "goed" behaald kan worden als  $H_s \leq 6,0$  m. Voor  $H_s > 6,0$  m wordt de kering verondersteld te falen ( $Z < 0$ ).

Wanneer de golfhoogte,  $H_s$ , kleiner is dan 6,0 m en de aanwezige laagdikte,  $D$ , groter is dan de hier berekende waarde  $D_{min}$ , dan geldt dat de kering tegen de golfbelasting bestand is. De grenstoestandfunctie is als volgt gedefinieerd:

$$Z = D - D_{min} \quad (4)$$

## Open steenasfalt

De algemene formule voor het bepalen van de minimaal benodigde laagdikte voor open steenasfalt wordt weergegeven met vergelijking (1). Evenals voor waterbouwasfaltbeton is de waarde van  $D_a$  een functie van golfhoogte  $H_s$  en taludhelling  $\tan \alpha$  en is de functiebeschrijving afhankelijk is van het type onderliggende laag (zand of klei). De functiebeschrijving is gebaseerd op fig. 3.2.2.7 uit de leidraad toetsen, en wordt ook toegepast in het softwarepakket PC-Ring (TNO-bouw, 2003a):

[3] open steenasfalt met ondergrond zand:

$$D_a = (-0.05785 + 0.025545 H_s + 0.026866 H_s^2) (-0.93087 + 5.108631 \tan \alpha + 0.060252 H_s) \quad (5)$$

[4] open steenasfalt met ondergrond klei:

$$D_a = (-0.1229 + 0.1782100 H_s + 0.03541978 H_s^2) (-1.36632 + 5.138124 \tan \alpha + 0.128026 H_s) \quad (6)$$

Afhankelijk van het filtertype moet voor de constanten  $C_3$  en  $C_4$  van vergelijking (1) de waarden uit Tabel 2.1 aangehouden worden.

Tabel 2.1 Parameters  $C_3$  en  $C_4$  voor open steenasfalt en twee verschillende typen filters.

	alle taluds	
	$C_3$	$C_4$
Geotextiel	0,2	0,025
Zandasfalt	0,1	0,025

Deze toetsingsmethode is alleen geldig wanneer de golfhoogte  $H_s$  kleiner dan 4,0 m is (fig. 3.2.2.7 uit de leidraad toetsen). Dit betekent dat alleen een score goed behaald kan worden als  $H_s \leq 4,0$  m. Voor  $H_s > 4,0$  m wordt de kering verondersteld te falen ( $Z < 0$ ).

Wanneer de aanwezige laagdikte groter is dan de hier berekende waarde, dan geldt dat de kering voldoende sterk is. De grenstoestandsfunctie is hetzelfde voor waterbouwasfaltbeton (vergelijking (4)).

### Aanname

De bovenstaande formules zijn gebaseerd op Figuur 3.2.2.6 en Figuur 3.2.2.7 van de Leidraad toetsen 1999. Bijbehorende constanten zijn afgeleid voor een beddingsconstante  $c_{zand} = 100$  MPa/m en een  $c_{klei} = 30$  MPa/m. Dit is in de meeste gevallen een veilige aanname.

## 2.3 Grenstoestandsfunctie voor grasbekleding

De methode voor het vaststellen van de betrouwbaarheid van grasbekledingen is gebaseerd op (WL, 2003). De stabiliteitsformules voor gras worden beschouwd voor de volgende belastingtypen:

- golfklappen op het buitentalud; en
- stroming op het buitentalud.

De stabiliteit van de kruin en het binnentalud is hierbij niet meegenomen, maar deze toetsing verloopt alsof het de buitenkruinlijn betreft en wijkt dus qua methode niet af. Er is onderscheid gemaakt tussen drie verschillende type grasmatten nl:

- goede grasmatt;
- matige grasmatt; en
- slechte grasmatt.

Voor de toetsing van grasbekledingen zijn de volgende invoergrootheden nodig:

$H_s$	:	significante golfhoogte op de teen van de constructie (m)
$T_p$	:	golfperiode behorende bij de piek van het spectrum (s)
$h$	:	waterstand (m + NAP)
$z_o$	:	niveau ondergrens van de te toetsen bekleding t.o.v. NAP (m)
$z_q$	:	oploophoogte behorende bij overslag debiet van $q = 0.1$ l/s/m op een oneindig lang talud t.o.v. SWL (m)

- $t_s$  : tijdstap van de berekening (uur)  
 $\tan \alpha$  : gemiddelde taludhelling bepaald tussen SWL +/-  $1.5H_s$ . Hierbij telt een eventuele berm niet mee bij het bepalen van de gemiddelde helling (-)  
*Type grasmat* : goed, matig, slecht

De toetsing start met een beoordeling of het stuk bekleding wel belast wordt en of er sprake is van golfklappen of alleen stroming door golfoploop en golfneerloop (fig. 4.7.2.1 LTV'99). Hierbij wordt het oplooptniveau  $z_q$  bepaald overeenkomstig de formuleringen geïmplementeerd voor het faalmechanisme golfoverslag (HKV, 2000, app. D) waarbij de exponentiële relatie tussen het overslagdebiet  $q$  en de kruinhoogte (cq. oplooptniveau  $z_q$ ) is omgekeerd.

- Als  $z_o > h + z_q \Rightarrow$  geen bijdrage aan beschadiging
- Als  $z_o < h \Rightarrow$  toetsing op golfklappen
- Als  $h \leq z_o \leq h + z_q \Rightarrow$  toetsing op stroming

Voor beide belastingen geldt dat de maximale belasting afhankelijk is van de belastingduur. Ten behoeve van de toetsing wordt deze relatie omgekeerd: uitgaande van de momentane belasting wordt de maximale belastingduur  $t_{s \max}$  bepaald. Vervolgens wordt de relatieve belastingduur  $t_s / t_{s \max}$  van de tijdstap  $t_s$  bepaald en gesommeerd over alle tijdstappen binnen de storm. Wanneer de gesommeerde relatieve belastingduur groter is dan 1 wordt de bekleding geacht te falen.

## Golfklappen

De toetsmethode voor golfklappen is gebaseerd op Figuur 4.7.2.2 uit de Leidraad toetsen. Deze figuur definieert een relatie tussen de maximaal toelaatbare golfhoogte en de maximale belastingduur ( $t_{sr} = t_s =$  stapgrootte in HYDRA-K). Deze relatie is afhankelijk van het type grasmat (goed, matig of slecht). Voor een belastingduur korter dan 5 uur geldt dat de maximale toelaatbare golfhoogte onafhankelijk is van de belasting duer  $t_s$ . Dit vertaalt zich in de conditie dat als de golfhoogte groter is dan de betreffende maximale golfhoogte, de bekleding bezwijkt:

$$\text{Als } H_s \geq \left[ C_1 \frac{1}{2} T_p^{-0.5} \tan \alpha^{-1} \right]^{0.75} \text{ dan } t_{s \max} = 0. \quad (7)$$

Waarbij  $C_1$  afhangt van het type grasmat volgens Tabel 2.2. In de berekening wordt  $t_s / t_{s \max}$  in deze situatie gelijk aan 2 gesteld.

Tabel 2.2 Parameter  $C_1$  voor golfklappen op grasbekledingen

	$C_1$
Goede grasmat	1
Matige grasmat	0,7
Slechte grasmat	0,2

Voor golfhoogtes kleiner dan een zekere waarde geeft Figuur 4.7.2.2 uit de Leidraad toetsen geen maximale belastingduur. De maximale belastingduur is in dat geval oneindig ( $t_{s \max} = \infty$ ) en de relatieve belastingduur  $t_s / t_{s \max}$  is dan dus gelijk 0:

$$\text{Als } H_s < \left[ 0.2 \frac{1}{2} T_p^{-0.5} \tan \alpha^{-1} \right]^{0.75} \text{ dan } t_s / t_{s \max} = 0. \quad (8)$$

Voor tussenliggende golfhoogtes geldt:

$$t_{s \max} = \left[ 2 H_s^{0.75} T_p^{0.5} \tan \alpha / C_2 \right]^{1/C_3} \quad (9)$$

In deze formule is de waarde van  $t_{s \max}$  in uur uitgedrukt. Afhankelijk van het type grasmat kunnen de waarden voor  $C_2$  en  $C_3$  de volgende waarden worden aangehouden:

Tabel 2.3 Parameters  $C_2$  en  $C_3$  voor golfklappen op grasbekledingen

	$C_2$	$C_3$
Goede grasmat	2.3	-0.52
Matige grasmat	1.6	-0.52
Slechte grasmat	0.2	0

## Stroming

Hier volgt het stappenplan voor het toetsen van een grasmat op stroming. De toetsing wordt uitgevoerd op een niveau  $z$  relatief t.o.v. de stilwaterlijn. Het meest kritieke niveau ligt zo laag mogelijk waar het oloopdebiet nog zo groot mogelijk is. Het niveau dat hier dan ook gebruikt wordt, is de ondergrens van het stuk van het talud dat met gras bekleed is:  $z = z_o - h$ . De berekening van de relatieve belastingduur bestaat uit een drietal stappen (zie fig. 4.7.2.3 uit de leidraad toetsen):

### Stap 1:

Bereken de rekensnelheid van de olooptong

$$v_r = 700 \frac{H_s}{T_p} \left( 0.085 - \frac{H_s}{\frac{g}{2\pi} T_p^2} \right) \sqrt{1 - \frac{z}{z_q} \tan \alpha} \quad (10)$$

hierin is  $g$  de zwaartekrachtsversnelling ( $\approx 9.81 \text{ m/s}^2$ ).

### Stap 2:

Bepaal met deze rekensnelheid de maximaal toelaatbare belastingduur (in uren):

$$t_{s \max} = C_1 v_r^{C_2} \quad (11)$$

In Tabel 2.4 staan de waarden voor de constanten  $C_1$  en  $C_2$  voor elk type grasmat (Goed, Matig en Slecht).

Tabel 2.4 Constanten  $C_1$  en  $C_2$  voor stroming op grasbekledingen

Grasmat	$C_1$	$C_2$
Goed	1560	-4,8
Matig	276	-4,2
Slecht	50	-3,6

### Stap 3

Bepaal de rekenwaarde van de belastingduur van de tijdstap  $t_s$ , waarvoor de berekening gemaakt wordt:

$$t_{sr} = \left(1 - \frac{z_o}{z_q}\right) t_s \quad (12)$$

Hierbij zijn de tijdstap  $t_s$  en de rekenwaarde  $t_{sr}$  voor de tijdstap beiden in uren.

### Cumulatieve graserosie

Uit zowel de toets op golfklappen als de toets op stromingcondities volgt een maximale belastingduur  $t_{s \max}$  welke gecombineerd met de tijdstap  $t_s$  resp. de rekenwaarde  $t_{sr}$  voor de tijdstap de relatieve belasting(duur) voor een enkele tijdstap geeft door het quotiënt te bepalen:

$$belasting = \frac{t_{sr}}{t_{s \max}} \quad (13)$$

Deze stappen moeten voor alle tijdstappen in de storm herhaald worden. De totale cumulatieve relatieve belasting wordt dan (zie VTV, 2003):

$$belasting = \sum \frac{t_{sr}}{t_{s \max}} \quad (14)$$

De grenstoestandsfunctie is gelijk aan:

$$Z = 1 - \sum \frac{t_{sr}}{t_{s \max}} \quad (15)$$

Dat wil zeggen: als de totale relatieve belastingduur groter is dan 1 treedt bezwijken van de grasmat op.

### Algemene opmerkingen m.b.t. grastoets

Het toetsniveau voor grasbekleding moet hoger liggen dan het niveau met een belastingkans van 1/10 per jaar (punten die onder dit niveau liggen vallen onder het dagelijks beheer). De kruin en het binnentalud worden getoetst alsof het de buitenkruinlijn is.

## 2.4 Grenstoestandsfunctie voor steenzetting

De methode voor het vaststellen van de betrouwbaarheid van steenzettingen is gebaseerd op figuur 2.2.4.5 en 2.2.2.3 van de leidraad Toetsen op Veiligheid (TAW, 1999). Het faalmechanisme kan het beste omschreven worden als “*instabiliteit van de toplaag van de steenzetting of lokale afschuiving, als gevolg van golven*”. Het faalmechanisme “*materiaaltransport*” wordt niet meegenomen omdat deze niet afhankelijk is van de golfhoogte, periode of waterstand. Bovendien is dit faalmechanisme meestal niet maatgevend.



Er worden 2 typen steenzettingen onderscheiden nl.:

- ongunstige steenzetting;
- normale steenzetting.

De basis voor de toetsing vormt de eenvoudige toetsmethode type c uit de leidraad, nl. 'ingeklemde stenen op een granulaire filter'. Van dit type zullen alleen de ongunstige constructies en de normale constructies beschouwd worden.

Andere type steenzettingen hebben een grenstoestandfunctie die sterk lijkt op die van de twee typen die hier gemodelleerd worden. Die bekledingstypen komen veel minder vaak voor in Nederland.

Voor de toetsing van steenzettingen zijn de volgende invoergrotheden nodig:

$H_s$	: significante golfhoogte op de teen van de constructie (m)
$h$	: waterstand (m + NAP)
$z_o$	: niveau ondergrens van de te toetsen bekleding (m + NAP)
$z_b$	: niveau bovengrens van de te toetsen bekleding (m + NAP)
$T_p$	: golfperiode behorende bij de piek van het spectrum (s)
$\tan \alpha$	: gemiddelde taludhelling bepaald tussen SWL +/- $1.5H_s$ . Hierbij telt een eventuele berm niet mee bij het bepalen van de gemiddelde helling (-)
$D$	: dikte van de toplaag (m)
$b$	: dikte van de filterlaag en kleilaag tezamen (m)
$D_{15}$	: karakteristieke korrelgrootte van het zand (m)
$\Delta$	: relatieve dichtheid $\Delta = (\rho_s - \rho_w) / \rho_w$ (-)
$\rho_w$	: soortelijke massa van water ( $\text{kg/m}^3$ )
$\rho_s$	: soortelijke massa van toplaag ( $\text{kg/m}^3$ )
$z_q$	: oploophoogte behorende bij overslag debiet van $q = 0.1$ l/s/m op een oneindig lang talud (m + NAP)
<i>type bekleding</i>	: normaal, ongunstig

De toetsingsmethode verloopt aan de hand van de volgende stappen:

#### Stap 0: afschuiving

Er treedt geen falen als gevolg van afschuiving op als voldaan wordt aan (fig. 2.2.2.3 LTV'99):

$$\Delta D + b > \min \left\{ 0,16 H_s^{0,2} T_p^{1,6} (\tan \alpha)^{0,8} ; 1,5 H_s \right\} - 1334 \cdot (1 - 1,19 \cdot \tan \alpha) D_{15} \sqrt{T_p} \quad (16)$$

Het toetsresultaat is dan goed en de toetsing dient te vervolgen met een toets op toplaagstabiliteit. Indien niet aan de voorwaarde wordt voldaan is het toetsresultaat twijfelachtig of onvoldoende; bij deze implementatie wordt de bekleding dan geacht te falen en hoeft de toets op toplaagstabiliteit niet meer uitgevoerd te worden.

#### Stap 1 t.a.v. toplaagstabiliteit (zie Steentoets 3.20)

$$\text{Definieer: } z_t = h - \min \left\{ 0,11 H_s \left( \frac{g T_p^2 \tan \alpha}{2\pi H_s} \right)^{0,8} ; 1,5 H_s \right\}$$

Als

- $z_t > z_b$   $\Rightarrow$  bekleding wordt niet belast.
- $h + \frac{1}{2} z_q < z_o$   $\Rightarrow$  bekleding faalt niet ( $Z > 0$ )
- $h < z_o$  en  $h + \frac{1}{2} z_q > z_o$   $\Rightarrow D = D/0.8 \Rightarrow$  toetsing uitvoeren (Stap 2)
- geen van de voorgaande gevallen  $\Rightarrow$  toetsing uitvoeren (Stap 2)

Hierbij wordt het oplooptniveau  $z_q$  bepaald overeenkomstig de formuleringen geïmplementeerd voor het faalmechanisme golfoverslag (HKV, 2000, app. D) waarbij de exponentiële relatie tussen het overslagdebiet  $q$  en de kruinhoogte (cq. oplooptniveau  $z_q$ ) is omgekeerd.

### Stap 2

Bepaal het stabiliteitsgetal,  $n_s$ , en de golfbrekerparameter,  $\xi_{op}$ , als volgt:

$$n_s = \frac{H_s}{\Delta D} \quad (17)$$

$$\xi_{op} = \frac{\tan \alpha}{\sqrt{\frac{H_s}{\frac{g}{2\pi} T^p}}} \quad (18)$$

### Stap 3

Bepaal de toetscriteria m.b.t. de grens tussen goed en twijfelachtig (fig. 2.2.4.5 LTV'99):

- Voor een *normale constructie* geldt m.b.t. de grens tussen goed en twijfelachtig:
  - voor  $0.6 < \xi_{op} \leq 2.0$   $\Rightarrow n_{s;onder} = 4.08 \xi_{op}^{-1.014}$
  - voor  $2.0 < \xi_{op} < 5.0$   $\Rightarrow n_{s;onder} = 11.0 \xi_{op}^{-4} + 0.03 \xi_{op} + 1.25$
- Voor een *ongunstige constructie* geldt m.b.t. de grens tussen goed en twijfelachtig:
  - voor  $0.6 < \xi_{op} \leq 2.0$   $\Rightarrow n_{s;onder} = 3.07 \xi_{op}^{-1.014}$
  - voor  $2.0 < \xi_{op} < 5.0$   $\Rightarrow n_{s;onder} = 6.5 \xi_{op}^{-4} + 0.02 \xi_{op} + 1.09$

De grenstoestandsfunctie tussen goed en twijfelachtig is als volgt (merk op dat  $n_{s;onder}$  een bovengrens is voor het gebied van  $n_s$  waar de bekleding goed is):

$$Z = n_{s;onder} - n_s \quad (19)$$

Eventueel kan ook een toetsing uitgevoerd worden m.b.t. de grens tussen twijfelachtig en onvoldoende. Bij de opzet van HYDRA-K is er echter voor gekozen om te toetsen op de grens tussen goed en twijfelachtig, waardoor onderstaande formules niet nodig zijn. Voor testdoeleinden kan wel gebruik gemaakt worden van deze grens omdat deze een iets afwijkend verloop heeft dan de grens tussen goed en twijfelachtig:

- Voor een *normale constructie* geldt m.b.t. de grens tussen twijfelachtig en onvoldoende:
  - voor  $0.6 < \xi_{op} \leq 2.1$   $\Rightarrow n_{s;boven} = 6.68 \xi_{op}^{-0.723}$
  - voor  $2.1 < \xi_{op} < 5.0$   $\Rightarrow n_{s;boven} = 12.0 \xi_{op}^{-1.5} + 1.5 \xi_{op} - 3.12$

- Voor een *ongunstige constructie* geldt m.b.t. de grens tussen twijfelachtig en onvoldoende:  
voor  $0.6 < \xi_{op} \leq 2.3$   $\Rightarrow n_{s; boven} = 3.07 \xi_{op}^{-1.014}$   
voor  $2.3 < \xi_{op} < 5.0$   $\Rightarrow n_{s; boven} = 13.8 \xi_{op}^{-4} + 0.26 \xi_{op} + 1.53$

De grenstoestandsfunctie tussen goed en twijfelachtig is als volgt:

$$Z = n_{s; onder} - n_s \quad (20)$$

## 3 Implementatie van de nieuwe functionaliteiten

### 3.1 Overwegingen en keuzes

#### 3.1.1 Omgaan met meerdere typen bekledingen op één dijkprofiel

In de bestaande versies van HYDRA-K wordt gerekend met één bekledingstype, nl. steenzetting (blokken). In het demonstratiemodel, daarentegen, behoren analyses met meerdere bekledingstypen op één dijkprofiel tot de mogelijkheden. Het demonstratiemodel kan tevens omgaan met dijkprofielen waarbij voor één type (bijvoorbeeld een steenzetting) er verschillende diktes aanwezig zijn.

In het nieuwe demonstratiemodel wordt de dijkdoorsnede door de gebruiker opgesplitst in (kleine) delen waarvoor het mogelijk is de eigenschappen van de bekleding in te voeren en aan te geven tussen welke niveaus deze bekleding aanwezig is. Zo kan men aangeven dat van NAP + 1,20 m tot NAP + 2,30 m er een normale steenzetting is met dikte van 0,30 m. en soortelijke massa van  $\rho = 2.350 \text{ kg/m}^3$ , dat die van NAP + 2,30 m tot NAP + 5,30 m. 0,40 m. dik is en dat vanaf NAP + 5,30 m er een goede grasmat aanwezig is. Het programma rekent dan voor elk van deze delen de faalkans uit. Dit betekent dat gedurende een faalkansberekening de Z-functie per wind/waterstand combinatie meerdere keren doorgerekend wordt.

Voor ontwerpberekeningen geldt eveneens dat meerdere niveaus van de dijk doorgerekend moeten worden. Met andere woorden: voor de verschillende bekledingstypen op een dijkprofiel moet afzonderlijk de gewenste eigenschappen bepaald worden opdat aan de gewenste faalfrequentie voldaan wordt. Probleem hierbij is dat de aldus bepaalde eigenschappen weliswaar aan de opgegeven faalfrequentie voldoen, maar de dijk als geheel in de regel niet. Daarom is verkozen om in het demonstratiemodel geen ontwerpberekeningen voor samengestelde dijkbekledingen te ondersteunen. Het is aan de gebruiker om met behulp van het model de afzonderlijke delen te toetsen. Dit is analoog aan de benaderingswijze die in de leidraad toetsen (TAW, 1999) gehanteerd wordt.

#### 3.1.2 Omgaan met het stormverloop

##### Probleemstelling

Onder invloed van een golfaanval kan een taludbekleding (gras, steen, asfalt) bezwijken en daarmee het falen van de gehele kering inleiden. De beschrijving van het gehele faalproces is uitermate complex. Bij een steenbekleding zullen meestal eerst enkele lokale beschadigingen optreden. Daarna krijgen de golven meer vat op de constructie en kan een

gat van enkele vierkante meters in de bekleding worden geslagen. Dit stadium wordt vervolgd door het aantasten van het onderliggende filter en kleilagen waarna tenslotte de zand- of kleikern aan erosie wordt blootgesteld.

In het programma PC-Ring is dit hele proces gemodelleerd en in de berekening opgenomen (TNO-bouw, 2003a). Uiteraard bevat dit proces een groot aantal vereenvoudigingen en schematiseringen. Eén van de belangrijkste vereenvoudigingen is dat gedaan wordt alsof de waterstand en golfhoogte zich gedurende de gehele storm op het maximale niveau bevinden. Verder wordt elke bekleding die op het talud aanwezig is geacht aan die belasting te worden blootgesteld. Tenslotte wordt het falen van de bekleding (gras uitgezonderd) geacht plaats te vinden direct bij het begin van de storm: de onderliggende lagen en kleikern krijgen bij falen van de bekleding dus de hele stormduur te verwerken.

## Afwegingen

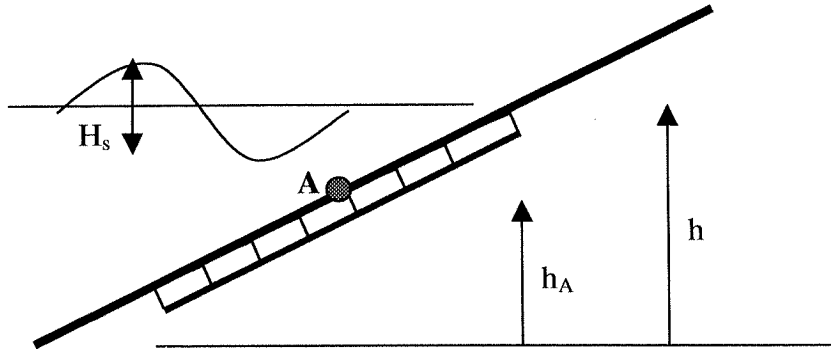
De vraag is nu welk model het beste in het demonstratiemodel kan worden ingevoerd. In vergelijking met bovengeschetste schematisering zal het programma worden beperkt tot de beschadiging van de top laag. Voor de mechanismebeschrijving wordt aangesloten bij de leidraad toetsing (TAW, 1999). Hiermee is dus een sterke vereenvoudiging bereikt. Aan de andere kant wordt wel rekening gehouden met het feit dat de storm een zeker verloop heeft en dat derhalve de meeste onderdelen van het talud maar gedurende een korte tijd en niet altijd bij maximale golfbelasting zullen worden aangevallen.

Uiteindelijk gaat het vervolgens om het uitrekenen van een toetspunt (of een aantal toetspunten) waarmee eenvoudige controles en ontwerpberekeningen kunnen worden uitgevoerd. Het is dus niet de bedoeling om via HYDRA-K een gedetailleerde of geavanceerde toetsing of ontwerp uit te voeren.

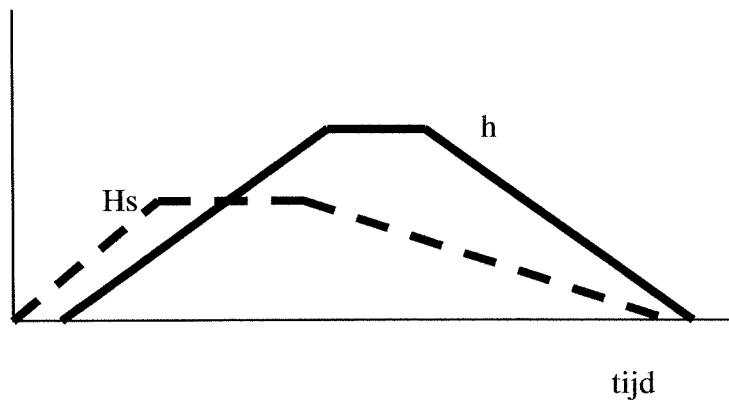
Bij een toetsing kan het eindresultaat voldoende, onvoldoende of twijfelachtig zijn. Bij “voldoende” of “onvoldoende” is inderdaad afdoende aangetoond dat geen of wel maatregelen nodig zijn. Bij het predikaat “twijfelachtig” is nog geen voldoende veiligheid aangetoond maar wordt de kans redelijk geacht dat nader onderzoek alsnog tot een voldoende resultaat zal leiden. Dit nader onderzoek kan van velerlei aard zijn. Het kan bijvoorbeeld zijn dat van een bepaalde bekleding de dikte niet bekend is en er voorlopig van een kleine waarde is uitgegaan. Nader onderzoek zou dan zekerheid kunnen verschaffen over de werkelijke dikte en daarmee uitsluitsel geven over het al dan niet voldoen van de bekleding. Ook kan mogelijk gedrag tijdens een recente storm gebruikt worden als een update van de berekening. Men kan echter ook denken aan het hanteren van betere en meer verfijnde modellen. In al deze gevallen kan men via simulaties vooraf nagaan of een dergelijke actie zinvol is. Als dat het geval is, is daarmee de status twijfelachtig gerechtvaardigd. Voor de procedure zelf heeft dit geen consequenties.

Beschouw nu een bepaald niveau van het talud (punt A in Figuur 3.1). De belasting heeft gedurende de storm een verloop als weergegeven in Figuur 3.2. Eerst bevindt het onderdeel zich ver boven de stilwaterlijn en is er in het geheel geen belasting. Vervolgens zullen de tongen van de hoogste golven de plaats A op het talud bereiken. Met het voortschrijden van de tijd zullen de intensiteit van de golven en de waterstand toenemen. Het stilwaterniveau

zal afhankelijk van de positie van A dit punt al dan niet passeren. In een later stadium van de storm kan eventueel de belastinghistorie in omgekeerde richting worden afgelegd.



Figuur 3.1 Talud onder golfaanval



Figuur 3.2 Verloop van waterstand en significante golfhoogte in de tijd

### Keuzes bij de implementatie in HYDRA-K

Een faalfrequentie-berekening in HYDRA-K bestaat grofweg uit de volgende vijf stappen:

**stap 1.** Inlezen van simultane waarnemingen van wind en waterstand.

**stap 2.** Selectie van de hoogste waterstanden tijdens elke stormgebeurtenis met de bijbehorende windsnelheid (deze stap wordt alleen uitgevoerd als de gebruiker de optie “hoogste waterstand” geselecteerd heeft. Indien de optie “ongunstigste belasting” geselecteerd is wordt de gehele meetreeks meegenomen).

Vervolgens worden voor elk van de geselecteerde simultane waarnemingen de volgende stappen uitgevoerd:

**stap 3.** Opschalen van de windsnelheid en de waterstand.

**stap 4.** Bepalen van de hydraulische belasting voor de teen van de dijk (waterstand, golfhoogte, -periode en -richting) op basis van de opgeschaalde waarden van windsnelheid en waterstand.

**stap 5.** Vaststellen of de hydraulische belasting groter is dan de weerstand van de kering, in welk geval de kering verondersteld wordt te falen.

Stap twee geeft aan dat er twee selectie-mogelijkheden zijn voor het selecteren van beschikbare simultane wind/waterstandmetingen. De wijze van selecteren is afhankelijk van de keuze van de gebruiker omtrent welke hydraulische belasting als representatief voor een stormgebeurtenis wordt beschouwd:

**Optie I:** de belasting op het moment dat de *waterstand* maximaal is; of

**Optie II:** de belasting op het moment dat de *belasting* maximaal is.

In het eerste geval worden uit de meetreeks eerst de hoogste waterstanden met bijbehorende windsnelheden geselecteerd en vervolgens de stappen 2 t/m 5 doorlopen. In het tweede geval worden de stappen 2 t/m 5 doorlopen voor alle beschikbare simultane metingen, waarna vervolgens per stormgebeurtenis het moment van maximale belasting wordt vastgesteld.

In het geval een berekening uitgevoerd wordt voor een dijkprofiel met verschillende bekledingstypen is optie I voor het bepalen van de faalkans van bekledingen onbruikbaar<sup>1</sup>. Immers, een stuk bekleding dat op een relatief laag niveau ligt, bevindt zich op het moment van de hoogste waterstand wellicht onder de stilwaterlijn waardoor bijvoorbeeld de belasting door golfklappen op dat moment nihil is. In de aanloop naar of de uitloop van de stormgebeurtenis kan de belasting door golfklappen mogelijk wel groot zijn op het moment dat het bewuste deel van het profiel zich boven de stilwaterlijn bevindt. De keuze voor optie I zou derhalve een grove onderschatting van de hydraulische belasting opleveren.

Voor alle bekledingstypes wordt daarom voorgesteld om het principe van optie II te volgen, d.w.z. voor elk uur in de storm wordt de hydraulische belasting bepaald en vervolgens wordt de maximale belasting per stormgebeurtenis bepaald. Dit maximum wordt per taludniveau bepaald. Bij grasbekleding is daarbij ook nog eens de belastingduur van belang. Optie I blijft alleen beschikbaar in het model voor dijken met uniforme bekledingstypen, en dan nog alleen voor faalmechanismen waar de stormduur geen rol van betekenis speelt.

Tot slot wordt opgemerkt dat de (gedwongen) keuze voor het principe van optie II zal leiden tot significant langere rekentijden.

### 3.1.3 Stormduur als additionele stochast

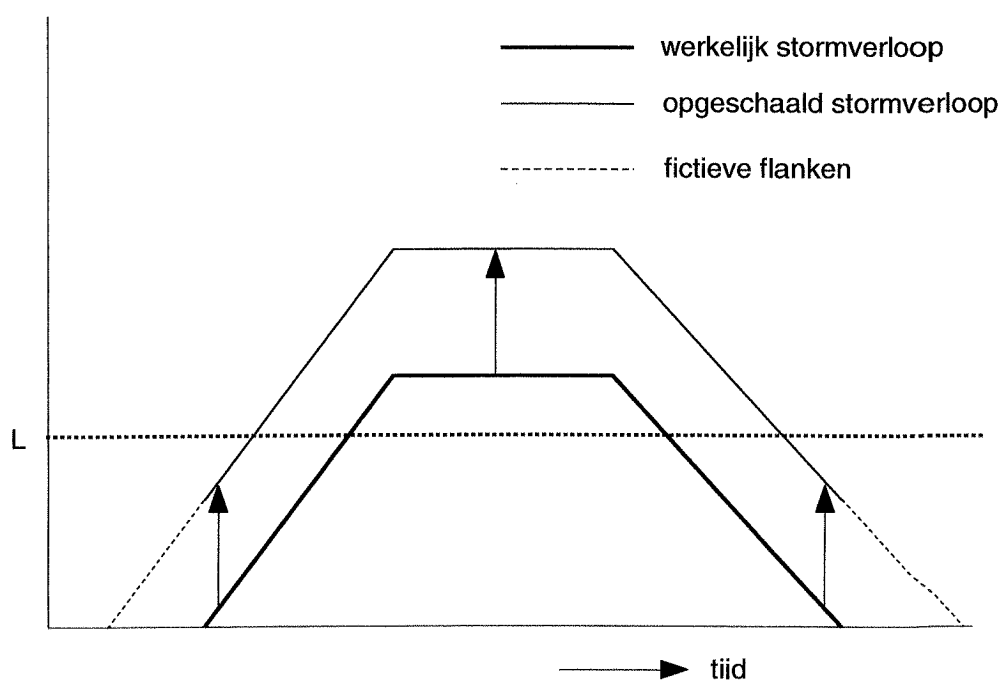
Bij grasbekleding is de duur van de hydraulische belasting (stormduur) één van de bepalende factoren. In feite is de stormduur daarmee op “verborgen” wijze verworpen tot één van de stochasten in het hydraulische belastingmodel. Met “verborgen” wordt bedoeld op het feit dat de duur niet expliciet is opgenomen in de methode de Haan, maar door het opschalen van de windsnelheid en de waterstand wel degelijk toeneemt en in feite dus ook opgeschaald wordt. Echter, het is niet gezegd dat deze wijze van opschalen van de stormduur zomaar is toegestaan. Immers, de overige twee stochasten (windsnelheid en waterstand) moeten ook voldoen aan bepaalde voorwaarden om de “methode de Haan” te mogen toepassen (nl. onderlinge asymptotische afhankelijkheid).

<sup>1</sup> voor het bepalen van faalkansen voor de faalmechanismen “golfoploop” en “golfoverslag” is optie I wel bruikbaar

Aan welke eisen exact voldaan moet worden hangt af van de wijze waarop de duur gemodelleerd wordt. Eén van de mogelijkheden is om een verdelingsfunctie voor de duur af te leiden en op basis daarvan de duur op te schalen in het standaard exponentiële vlak, op dezelfde wijze als de windsnelheid en de waterstand. In dat geval dient de duur asymptotisch afhankelijk te zijn van de wind en de waterstand. Een andere mogelijkheid is om de stormduur af te leiden uit de hydraulische belasting voor de dijk, die volgt uit de opgeschaalde windsnelheid en waterstand (zoals eerder geschetst). Door opschaling van wind en waterstand neemt de stormduur automatisch toe (zie Figuur 3.3). Uit data-analyse zal moeten blijken of bij een toename van de windsnelheid en de waterstand de stormduur in werkelijkheid gemiddeld evenveel toeneemt als weergegeven in deze figuur.

In de offertefase is met de opdrachtgever overeengekomen dat faalmechanismen en bekledingstypen die niet binnen het concept van HYDRA-K passen niet geïmplementeerd worden in het demonstratiemodel van fase 2 en dat een work-around wordt beschreven in fase 4. De term “binnen het concept passen” is nogal subjectief, maar afhankelijk van de uitkomsten van bovenstaande data-analyses kan gesteld worden dat het belastingtype “stroming op het buitentalud” bij grasbekleding vanwege de invloed van de duur voor deze kwalificatie in aanmerking komt. Desondanks wordt voorgesteld om dit belastingstype toch te implementeren in het demonstratiemodel. Zolang benadrukt wordt dat in dat geval *wellicht* niet aan de vereiste voorwaarden van de gebruikte methode wordt voldaan is dat niet erg. Het gaat hier immers om een demonstratiemodel en niet om een “echte” versie van HYDRA-K. In ieder geval wordt in fase 4 onderzocht of het opstellen van een work-around voor het bekledingstype gras noodzakelijk is.

Tot slot wordt opgemerkt dat de berekening van het ontwerppunt voor grasbekleding vanwege de invloed van de stormduur niet is opgenomen in het demonstratiemodel.



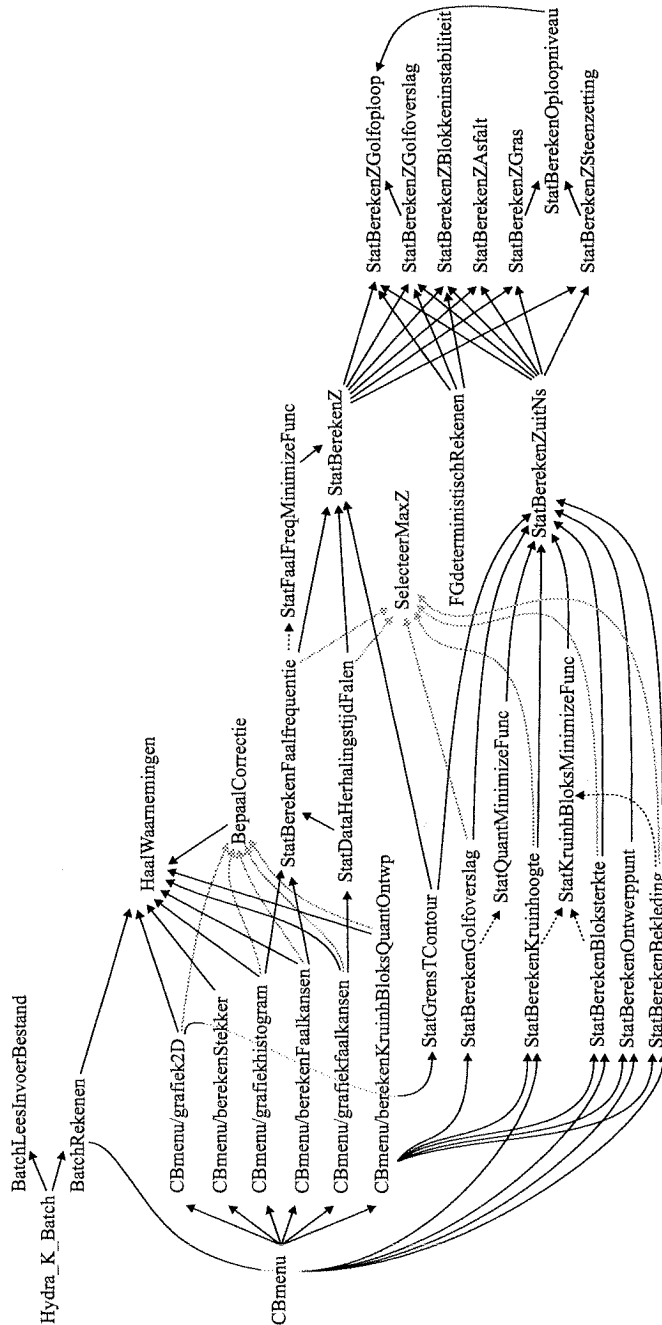
Figuur 3.3 Schematische weergave van het opschalen van het stormverloop in de tijd, hetgeen leidt tot een toename in de overschrijdingsduur van het niveau L



## 3.2 Beschrijving van de implementatie

### 3.2.1 Inleiding

Deze sectie beschrijft de aanpassingen van de HYDRA-K code die nodig waren om de faalmechanismen en bekledingstypes, zoals beschreven in hoofdstuk 2, binnen de demonstratieversie beschikbaar te krijgen. Tevens worden optredende complicaties vermeld. Ten behoeve van de inzichtelijkheid geeft Figuur 3.1 een overzicht van de belangrijkste routines binnen het model en de wijze waarop ze elkaar aanroepen.



Figuur 3.4 Schematische weergave van de samenhang van de belangrijkste routines. De pijlen in de figuur representeren de aanroep van de module waarnaar zij verwijzen.

### 3.2.2 Z-functies

De belangrijkste stap binnen deze fase van het project is het operationaliseren van de Z-functies voor asfalt, gras en steenzettingen. Hiertoe zijn drie routines toegevoegd: *StatBerekenZAsfalt*, *StatBerekenZGras* en *StatBerekenZSteenzetting*. De routines volgen de formuleringen voor de toetsing van de grenstoestandsfuncties zoals omschreven in hoofdstuk 2.

Een tweede belangrijke aanpassing in vergelijking met voorgaande versies van HYDRA-K is de mogelijkheid om dijken met meerdere bekledingstypes te toetsen. Samengestelde dijken worden daartoe binnen de programmatuur in stukken opgedeeld en elk deel wordt aan de toets onderworpen. In alle relevante routines wordt eerst gecontroleerd of het beschouwde stuk van de bekleding wel belast wordt. Desondanks wordt ook in het geval de bekledingen niet belast wordt toch alle vervolgberekeningen uitgevoerd om de efficiëntie van de gevectoriseerde aanpak in MATLAB te kunnen handhaven. Een belangrijke rol is daarbij weggelegd voor de index-vector “resterend” die aangeeft over welke van de simultane waarnemingen nog geen definitief uitsluitsel bestaat over het toetsingsresultaat.

In het geval van gras wordt er eerst per tijdstap de relatieve belastingduur bepaald; hierbij worden alle waarnemingen nog onafhankelijk verwerkt. Vervolgens wordt per stormgebeurtenis de cumulatieve relatieve belastingduur bepaald, hetgeen de uiteindelijke waarde voor de belasting geeft (zie paragraaf 2.3).

#### **Opmerkingen:**

- Naast de nieuwe bekledingstypen is ook de mogelijkheid aangehouden om het reeds in HYDRA-K bestaande bekledingstype “*blokken*” te selecteren, ondanks de grote overlap met het nieuw ingebouwde bekledingstype “*steen-zetting*”. Dit is gedaan om de code makkelijker te testen op interne consistentie van de applicatie zonder tegen verschillen in bekledingstype aan te lopen. De oude routine “*StatBerekenZBlokkeninstabiliteit*” routine is daarbij zo veel mogelijk intact gelaten. Deze routine controleert daarom niet op de grenzen van de bekleding en voert de toets uit op basis van een uniforme blokken bekleding van de hele dijk ongeacht de door de gebruiker gespecificeerde range. Het bekledingstype blokken dient dan ook niet in combinatie met andere bekledingen gebruikt te worden!
- Het bekledingstype blokken is identiek aan de situatie met steenzetting mits de laagtedikte van de filter- en kleilaag bij laatstgenoemde voldoende dik wordt gekozen. In dat geval is falen als gevolg van lokale afschuiving bij steenzetting namelijk niet mogelijk, met als gevolg dat de steenzetting alleen op instabiliteit van de toplaag getoetst wordt. Desondanks kunnen ook bij grote (theoretisch oneindige) laagtedikte van de filter- en kleilaag kleine verschillen optreden ten gevolge van een iets andere afhandeling van de hellingshoek in de buurt van de berm.
- In het geval van een samengestelde dijkbekleding vindt het aanroepen van de afzonderlijke routines ter berekening van de Z-functies en de verwerking van de daaruit voortkomende resultaten plaats op het niveau waarop ook de andere Z-functies worden aangeroepen, zoals in *StatBerekenZ*. Zie ook de volgende paragraaf.
- De routine *StatBerekenZBlokinstabiliteit* is uitgebreid met de optie om de blokgegevens (blokdikte en dichtheid) te kunnen lezen uit de datastructuur welke voor de samengestelde profielen is gedefinieerd.

- Ten behoeve van de bepaling van de locale effectieve helling van het dijktaalud is een routine *BepaalHelling* toegevoegd in de directory *Statprivate*.
- Als onderdeel van de Z-functies van steenzettingen en gras moet het oplooppniveau  $z_q$  behorende bij een overslag debiet van  $q = 0.1$  l/s bepaald worden ten opzichte van SWL. De methode die hiervoor gebruikt wordt vertoont grote overeenkomsten met de bestaande routine *StatBerekenZGolfoverslag*, waarin het overslagdebet over een gegeven dijkprofiel berekend wordt. Het belangrijkste verschil is dat in *StatBerekenZGolfoverslag* het overslagdebet voor een *gegeven* kruinhoogte wordt bepaald, terwijl nu de omgekeerde (inverse) relatie nodig is. Deze omgekeerde relatie is op analytische wijze afgeleid en geïmplementeerd in een nieuwe functie *StatBerekenOplooppniveau*.

### 3.2.3 Aanroep van Z-functies

De Z-functies worden aangeroepen vanuit de routines: *StatBerekenZ* en *StatBerekenZuitNs*. De Z-functies voor de nieuwe bekledingen zijn toegevoegd aan deze twee routines. Om dijken met meerdere bekledingstypes te kunnen toetsen is het aanroepen van de Z-functies gewijzigd in vergelijking met de bestaande versies van HYDRA-K. Een dijk wordt verondersteld opgedeeld te zijn in één of meerdere delen met uniforme bekleding en elk van deze delen wordt in eerste instantie apart getoetst. Het volgende stappenplan is gehanteerd:

1. Voor elk profiel worden de stappen 2 t/m 4 uitgevoerd (bestaande lus)
2. Voor elk deel van het dijkprofiel wordt stap 3 uitgevoerd (nieuwe lus)
3. Aanroep Z functie
4. Bepalen minimale Z waarde over alle delen van de bekleding (en evt. bepaal bijbehorende q waarde in *StatBerekenZuitNs*)
5. Bepalen van de minimale Z-waarde

#### Opmerkingen:

- Stap 5 impliceert dat de dijk faalt gedurende een stormgebeurtenis indien tenminste één van de delen faalt. De leidraad toetsen (TAW, 1999) doet echter geen uitspraken over faalkansen van dijken die uit meerdere bekledingen bestaan. Volgens de leidraad moeten de afzonderlijke delen ook afzonderlijk getoetst worden en zolang elk van deze delen aan de gestelde eisen voldoet is voor de dijk als geheel voldoende. Om de berekeningen met HYDRA-K zoveel mogelijk te laten stroken met de leidraad, wordt daarom aangeraden om per berekening slechts één bekledingstype door te rekenen. Voor dijken met bijvoorbeeld drie verschillende bekledingstypen impliceert dit dat in totaal drie berekeningen (ontwerpberekeningen of faalkansberekeningen) uitgevoerd moeten worden.
- De lus voor de samengestelde dijkbekleding zou eventueel in een aparte routine gezet kunnen worden, maar gezien detailverschillen in de aanroepen van de Z-functies tussen *StatBerekenZ* en *StatBerekenZ* levert dat niet veel code-besparing op.
- In HYDRA-K worden de Z-functies van de faalmechanismen allen ook aangeroepen vanuit de routine *FGdeterministischRekenen*, (om deterministische berekeningen uit te voeren). In het demonstratiemodel is er voor gekozen om deze routine (vooralsnog) niet uit te breiden met de Z-functies voor de nieuwe bekledingen omdat dat niets toevoegt aan de doelstelling van de haalbaarheidsstudie. In het geval deze uitbreiding in de toekomst alsnog gerealiseerd wordt dient eerst een aantal keuzes gemaakt te worden ten

aanzien van de functionaliteit van samengestelde dijkbekledingen (nl. moeten de deterministische berekeningen voor elk afzonderlijk onderdeel uitgevoerd worden of voor de dijk als geheel?) en stormduurafhankelijke belastingen (nl.: moet de stormduur als extra invoerparameter dienen of juist als uitvoer?).

### 3.2.4 Faalfrequentie-berekeningen

Ten behoeve van faalfrequentie-berekeningen zijn de routines *StatDataHerhalingstijdFalen*, *StatBerekenFaalfrequentie* en *StatFaalFreqMinimizeFunc* aangepast op de volgende punten:

- De *WaarnStormIndex* vector wordt doorgegeven aan routine *StatBerekenZ*.
- Voor stormduurafhankelijke belastingen wordt vanuit routine *StatBerekenFaalfrequentie* niet meer alleen de ongunstigste waarneming in de 50<sup>ste</sup> (of, meer algemeen, de  $\kappa^e$ ) storm aan de optimalisatieroutine *StatFaalFreqMinimizeFunc* doorgegeven, maar de hele storm.
- De te minimaliseren functie  $z^*z$  is vervangen door  $\min(z)^2$ . Deze functie is nodig op het moment dat een hele storm wordt doorgegeven in plaats van een enkele waarneming.

De juiste bekledingsparameters worden door de functies *grafiekfaalkansen*, *berekenenFaalkansen* en *grafiekhistogram* aangeleverd. Zie paragraaf 3.2.7 voor een beschrijving van de aanpassingen in de subfuncties van *CBmenu*.

### 3.2.5 Ontwerpberekeningen

Centraal in de batch berekeningen staan de ontwerpberekeningen van kruinhoogte en bekledingskenmerken. Deze optie is ook via de user-interface aan te roepen vanuit het menu 'Rekenen' onder de naam 'Ontwerpberekening...' (voorheen 'Kruinhoogte/bloksterkte'). Deze functionaliteit maakt gebruik van de routines: *StatBerekenGolfoverslag*, *StatBerekenKruinhoogte*, *StatBerekenBloksterkte* en *StatBerekenOntwerppunt* en een tweetal hulpfuncties t.b.v. optimalisatie. Tabel 3.1 beschrijft de grootheden die in de ontwerpberekeningen worden geoptimaliseerd.

Tabel 3.1 te optimaliseren grootheden voor de verschillende bekledingstypes

bekledingstype	te optimaliseren grootheid	bereik
asfalt:	de laagdikte van het asfalt	1 cm - 50 m
blokken:	de bloksterkte (product van blokdikte en dichtheid)	1 cm - 50 m
gras:	de kwaliteit van de grasmat	slecht, matig, goed
steenzetting	de bloksterkte (product van blokdikte en dichtheid)	1 cm - 50 m

De bestaande routines zijn alleen aangepast met betrekking tot de aanroep van *StatBerekenZuitNs* en de hulpfunctie *StatKruinhBlokMinimizeFunc*, waaraan in beide gevallen de vector *WaarnStormIndex* is toegevoegd. Aan de laatstgenoemde hulpfunctie is tevens een optie toegevoegd ter ondersteuning van het ontwerp van de nieuwe typen bekleding. Ten behoeve van het ontwerp van asfalt, gras en steenzetting bekledingen is een routine *StatBerekenBekleding* toegevoegd analoog aan de reeds bestaande routine voor blokkeninstabiliteit.

**Opmerking:**

- Uit de tabel valt op te maken dat in het geval van gras de te optimaliseren grootheid discreet is. Voor discrete grootheden worden alle situaties getest, beginnend bij de slechtste kwaliteit. De minimale graskwaliteit die voldoet aan de veiligheidseis wordt als optimaal ontwerp weergegeven.

### 3.2.6 Algemene routines

Van de routines en scripts in de general directory zijn de volgende veranderd: *GlobalConstants*, *InitialiseerGlobalConstants* en *PrintWaarden*. De eerste twee modules zijn om administratieve redenen aangepast, d.w.z. de constante *FMALGEMINSTAB* (=4) is gedefinieerd welke de nieuwe optie met de algemene instabiliteit van bekledingen representeert.

**Opmerking:** Op de meeste plaatsen in de code wordt gewoon gebruik gemaakt van de faalmechanisme identificatienummers (1 t/m 4) zonder gebruik te maken van deze globale constanten. Deze constanten moeten eigenlijk consistent door de hele code worden gebruikt of worden verwijderd.

De routine *PrintWaarden* is uitgebreid met een aanroep van de nieuwe functie *PrintBekleding* die er voor zorgt dat de bekleding wordt weergegeven op het scherm en weggeschreven naar de uitvoerbestanden.

Tot slot zijn aan de general directory twee routines *MeerdereBekledingen* en *StormDuurAfhankelijk* toegevoegd welke gebruikt kunnen worden om te testen of een gegeven dijkbekleding uit meerdere segmenten met verschillende bekledingstypen en/of gelijke bekledingstypen met verschillende parameters bestaat, respectievelijk om te testen of de belasting in een van de relevante Z-functies stormduurafhankelijk is (nu nog effectief een check op de aanwezigheid van gras op het talud).

### 3.2.7 User interface

Aanpassingen in de user interface functies zijn aangebracht in de volgende bestanden: *CBmenu*, *CBfaalmechanisme*, *FGhydra\_k* en *FGkruinhoogteBlokdikte*. De wijziging in de laatste routine betreft slechts het doorvoeren van de naamswijziging van 'Kruinhoogte/bloksterkte' in 'Ontwerpberekening'.

De aanpassingen in *FGhydra\_k* betreffen dezelfde naamswijziging, het toevoegen van het nieuwe faalmechanisme 'algemene instabiliteit' met bijbehorende knop om de parameters te wijzigen, en een interne functie voor het ophalen van de bekledingsgegevens bij interactief gebruik.

De wijzigingen in *CBfaalmechanisme* betreffen de afhandeling van de zichtbaarheid van de toegevoegde knop. De wijzigingen in *CBmenu* betreffen het afvangen van menu commando's met duidelijke dialoogvensters wanneer de betreffende operatie niet uitgevoerd kan worden voor de door de gebruiker geselecteerde combinatie van parameters. Zo is de optie 'tweedimensionale grafieken' niet beschikbaar voor stormduurafhankelijke belastingen (d.w.z. bekleding met gras). Ten slotte is de routine *CBbekleding* toegevoegd om de parameters van één of meerdere bekledingen op het dijktalud interactief te kunnen wijzigen.

### 3.2.8 Batch routines

De aanpassingen van de batch routines betreffen de functies *BatchLeesInvoerBestand* en *BatchRekenen*. In de laatstgenoemde functie is ondersteuning opgenomen voor het nieuwe faalmechanisme 'algemene instabiliteit'. Ten behoeve van de algemene instabiliteit moeten de kenmerken van de dijkbekleding worden ingevoerd. De gegevens ten behoeve van de blokkeninstabiliteit komen uit de database.

### 3.2.9 Overige aanpassingen aan de code

Ten slotte zijn er nog twee andere aanpassingen in de code uitgevoerd die de werking van het programma versoepelen:

1. In de routine *ControleerRand2001Data* is een MATLAB prod commando werkend op het resultaat van een logische operatie vervangen door een MATLAB all commando. Dit geeft exact hetzelfde resultaat maar de tweede implementatie is duidelijker en is compatible met MATLAB 6.5.
2. De routine *StatWeibullInv* is versneld door de permute operatie op de vier grote matrices omega, mu, sigma en alpha te vervangen door dezelfde operatie op slechts twee matrices *OverschrFreq* en *a* (de eerstgenoemde is significant kleiner). Dit scheelt een factor twee in rekestijd, terwijl de module niets van haar functionaliteit heeft verloren.

## 4 Beknopte gebruikershandleiding voor de nieuwe faalmechanismen

### 4.1 Inleiding

Voor de gebruiker die bekend is met voorgaande versies van HYDRA-K (en we gaan er in dit hoofdstuk vanuit dat de lezer tot die groep behoort) hoeft het gebruik van het demonstratiemodel nauwelijks uitleg. Daarom zijn de aanpassingen in de user-interface en de batch-mode in dit hoofdstuk beknopt beschreven.

### 4.2 User interface

Het hoofdscherm in de user-interface is uitgebreid met één extra faalmechanisme (zie Figuur 4.1) genaamd "*Algemene instabiliteit*". Onder deze noemer vallen de nieuwe faalmechanismen voor bekledingstypen gras, steenzetting en asfalt. Zodra de gebruiker dit faalmechanisme selecteert verschijnt een balk in beeld met daarop de tekst "wijzig bekleding" (zie Figuur 4.1). Zodra deze aangeklikt wordt met de muis verschijnt een nieuw scherm in beeld, waarin de bekleding van de dijk gedefinieerd en/of aangepast kan worden (zie Figuur 4.2). Dit scherm bestaat uit twee delen:

1. aan de linker zijde zijn alle knoppen en edit vensters geplaatst.
2. aan de rechter zijde is het geselecteerde profiel weergegeven met daarop aangegeven met een rode lijn de locatie van het beschouwde bekledingssegment.

De gebruiker kan aan de linkerkzijde de parameters van de bekleding, het type bekleding en de locatie van het bekledingssegment wijzigen. Tevens is het mogelijk om de bekledingssegmenten toe te voegen en de totale opbouw van de dijkbekleding naar een bestand weg te schrijven. De optie van het wegschrijven naar een bestand is in de eerste plaats toegevoegd om later eenvoudig nieuwe berekeningen met hetzelfde profiel uit te kunnen voeren. Daartoe kan het opgeslagen bestand automatisch ingelezen worden. Een bijkomende functie van het wegschrijven van de gegevens van het dijkprofiel is dat deze als invoer van de batchberekeningen vereist zijn (zie volgende paragraaf).

**Opmerkingen:**

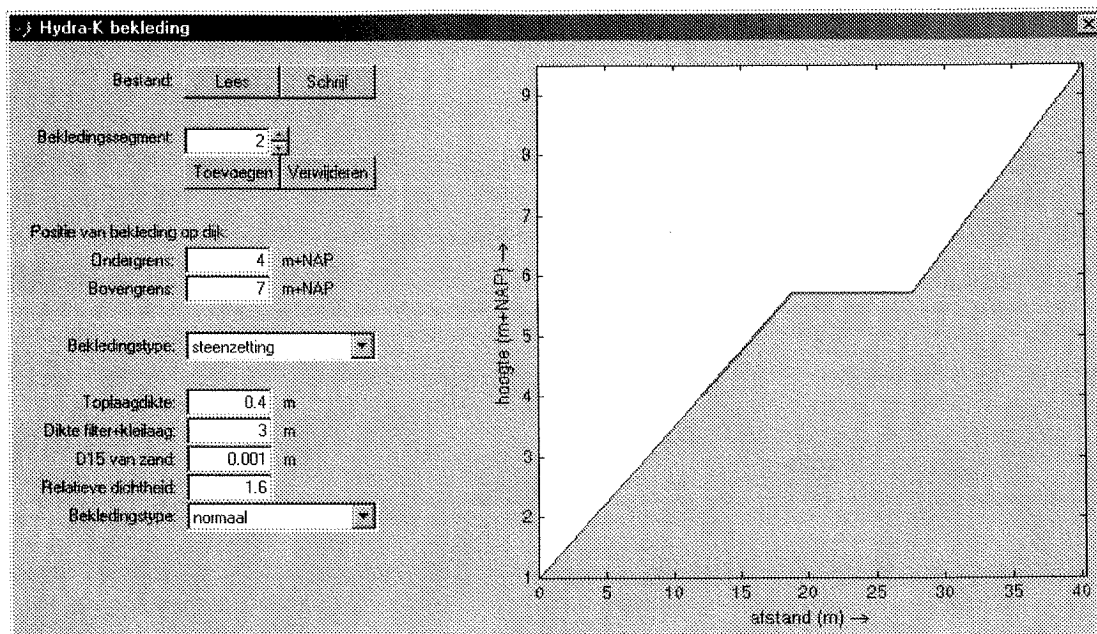
- HYDRA-K biedt de mogelijkheid om faalkansen voor dijken met meerdere bekledingstypen door te rekenen. De leidraad toetsen (TAW, 1999) doet echter geen uitspraken over faalkansen van dijken die uit meerdere bekledingen bestaan. Volgens de leidraad moeten de afzonderlijke delen ook afzonderlijk getoetst worden en zolang elk van deze delen aan de gestelde eisen voldoet is voor de dijk als geheel voldoende. Om de berekeningen met HYDRA-K zoveel mogelijk te laten stroken met de leidraad, wordt daarom aangeraden om per berekening slechts één bekledingstype door te rekenen. Voor dijken met bijvoorbeeld drie verschillende bekledingstypen impliceert dit dat in totaal drie berekeningen (ontwerpberekeningen of faalkansberekeningen) uitgevoerd moeten worden.
- In het geval van tijdsduurafhankelijke belasting (d.w.z. in het geval van grasbekleding) is het ontwerp punt niet gedefinieerd. De bepaling van het ontwerp punt wordt in dat geval overgeslagen.
- Indien de gebruiker de *combinatie* van standaard profielen heeft geselecteerd, zal het eerste profiel worden weergegeven in de interface.
- In deze eerste haalbaarheidsversie van het programma dient de gebruiker er zelf zorg voor te dragen dat de verschillende delen van de dijkbekleding op elkaar aansluiten en elkaar niet overlappen, d.w.z. er zijn geen waarschuwingen ingebouwd.

The screenshot shows the HYDRA-K software interface with the following parameters and options:

- Regio:** Kust noord
- Locatie:** X=104548, Y=527420 (<geïmporteerd)
- Dijkprofiel:** sp1
- Orientatie:** 281.18 graden
- Ruwheid:** 1
- Hoogte teen:** 0 m+NAP
- Helling laag:** 0.25
- Hoogte berm:** 5.13 m+NAP
- Helling berm:** 0
- Hoogte kruin:** 12.75 m+NAP
- Helling hoog:** 0.33333
- Breedte berm:** 13 m
- Aantal verschoven waarnemingen in het faalgebied:** 50
- Faalmechanisme:**
  - Golfploop
  - Golfoverslag
  - Blokkerinstabiliteit
  - Algemene instabiliteit
- Wijzig bekleding** (button)

Figuur 4.1 Bovenste deel van het hoofdscherm van de user-interface van HYDRA-K





Figuur 4.2 De user interface voor het wijzigen van de dijkbekleding.

### 4.3 Batchberekeningen

In verband met het karakter van deze studie is ervoor gekozen om de parameters van de dijkbekleding door de gebruiker in een apart bestand (dus niet uit de database) te laten specificeren. Zoals eerder gezegd dient dit bestand middels de user interface aangemaakt te worden. De aansturing van een batchbestand gebeurt nog steeds via een invoerbestand, waarvan onderstaand en voorbeeld gegeven is:

Regio = Kust zuid  
 Kappa = 50  
 Faalmechanisme = AL  
 Herhalingsstijd = 10 30 100 300 1000 2000 4000 10000  
 Profiel = ST,sp1  
 Afhankelijkheid = WAARnemingen, HOOGste waterstand  
 Stromingscorrectie = NIET  
 Vaste stilwaterlijn = NIET  
 Dijkopbouwbestand = mijndijk.bekleding  
 Uitvoerbestand = D:\test\test.txt

In vergelijking met voorgaande versies van HYDRA-K zijn twee dingen veranderd:

- Het faalmechanisme ‘algemene instabiliteit’ is toegevoegd als optie en kan worden geselecteerd door AL te specificeren bij het keyword *Faalmechanisme*.
- De opbouw van de dijkbekleding dient gespecificeerd te worden door middel van het keyword *Dijkopbouwbestand*, bijvoorbeeld: *Dijkopbouwbestand = mijndijk.bekleding* waarbij het bestand *mijndijk.bekleding* gemaakt is met de user interface.



## 5 Testberekeningen

### 5.1 Inleiding

Voor drie verschillende locaties (zie Tabel 5.1) is een aantal testberekeningen uitgevoerd met het nieuwe demonstratiemodel. Bij de berekeningen is aangenomen dat de dijken een zogenaamd “standaard profiel” hebben, waarvan Tabel 5.2 de gegevens bevat. Figuur 5.1 toont een schematische weergave van dit standaardprofiel.

In eerste instantie zijn testberekeningen uitgevoerd voor dijkprofielen met een uniform bekledingstype. Het voornaamste doel van deze berekeningen is om de nieuw geprogrammeerde rekenregels in het model op consistentie te testen. Vervolgens is een aantal berekeningen uitgevoerd voor *delen* van dijkprofielen om te testen in hoeverre daar problemen ontstaan die bij het doorrekenen van de gehele dijk niet aan het licht komen.

Tabel 5.1 Locaties waarvoor testberekeningen zijn uitgevoerd.

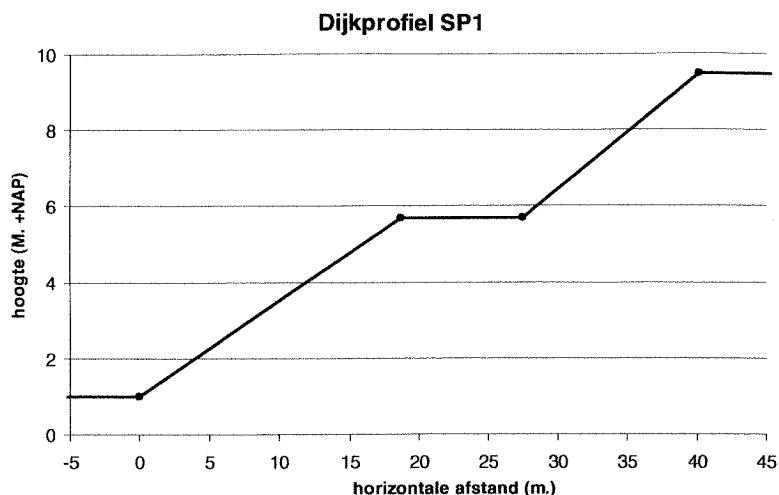
nummer	locatie	x-coördinaat	y-coördinaat	deelgebied
1	Vlissingen	29571	384983	Westerschelde
2	Den Helder	111752	553252	Waddenzee West
3	Petten	104548	527420	gesloten kust Noord

Tabel 5.2 Parameters van het standaard profiel.

parameter	waarde
Hoogte teen:	1 m+NAP
Hoogte berm:	5.7 m+NAP
Hoogte kruin:	9.5 m+NAP
Breedte berm:	8.75 m
Ruwheid:	1
Helling laag:	0.25
Helling berm:	0
Helling hoog:	0.29992

### 5.2 Uniforme bekleding, optie hoogste waterstand

In de eerste testberekeningen is voor de dijken op de drie locaties een uniform bekledingstype aangenomen. Als selectiecriteria voor de simultane waarnemingen is de optie “hoogste waterstand” gekozen, d.w.z. de belasting wordt bepaald voor het moment in de storm waarop de waterstand maximaal is. Met de keuze voor deze optie wordt de rekentijd aanzienlijk verkort. Voor in totaal 14 verschillende bekledingstypen (zie Tabel 5.3) zijn ontwerpberekeningen uitgevoerd bij herhalingstijden van 10, 30, 100, 300, 1.000, 2.000, 4.000 en 10.000 jaar.



Figuur 5.1 Profiel waarvoor de berekeningen zijn uitgevoerd.

Tabel 5.3 Bekledingstypen waarvoor berekeningen zijn uitgevoerd met de optie “hoogste waterstand”.

1.	Blokken
2.	Normale steenzetting met oneindig dikke klei + filterlaag
3.	Normale steenzetting met klei+filterlaag van 1 m.
4.	Normale steenzetting met klei+filterlaag van 0.1 m.
5.	Ongunstige steenzetting met klei+filterlaag van 1 m.
6.	Ongunstige steenzetting met klei+filterlaag van 0.1 m.
7.	Waterbouwasfaltbeton met filtertype geotextiel, ondergrond klei
8.	Waterbouwasfaltbeton met filtertype zandasfalt, ondergrond klei
9.	Open steenasfalt met filtertype geotextiel, ondergrond klei
10.	Open steenasfalt met filtertype zandasfalt, ondergrond klei
11.	Waterbouwasfaltbeton met filtertype geotextiel, ondergrond zand
12.	Waterbouwasfaltbeton met filtertype zandasfalt, ondergrond zand
13.	Open steenasfalt met filtertype geotextiel, ondergrond zand
14.	Open steenasfalt met filtertype zandasfalt, ondergrond zand

Opgemerkt wordt dat in deze exercitie geen grasbekleding is doorgerekend, omdat voor gras de duur van de storm een rol speelt en de optie “hoogste waterstand” derhalve niet bruikbaar is.

Tabel 5.5 bevat de resultaten van de beschreven exercities. Onderstaand beschrijven we de belangrijkste bevindingen.

[1] De resultaten van de eerste twee berekeningen zijn zoals verwacht nagenoeg gelijk. De klei+filterlaag bij steenzetting (berekening 2) is namelijk oneindig dik gekozen, zodat falen als gevolg van lokale afschuiving niet op kan treden. Het gevolg is dat de steenzetting alleen op instabiliteit van de toplaag getoetst wordt en dit faalmechanisme is vergelijkbaar met het faalmechanisme blokkenstabiliteit (berekening 1), hetgeen blijkt uit de resultaten. De kleine verschillen worden veroorzaakt door een iets andere afhandeling van de helling in de buurt van de berm. Opgemerkt wordt dat het faalmechanisme blokkeninstabiliteit al bestond in eerdere versies van HYDRA-K en onveranderd is gelaten in het huidige demonstratiemodel.

[2] De drie berekeningen voor normale steenzetting (2, 3 en 4) leveren exact dezelfde resultaten op, ondanks verschillend gekozen diktes van de klei- + filterlaag. Daaruit blijkt

dat het faalmechanisme “lokale afschuiving” een ondergeschikte rol speelt onder de doorgerekende omstandigheden.

[3] De twee berekeningen voor ongunstige steenzetting (5 en 6) leveren eveneens exact dezelfde uitkomsten, ondanks verschillend gekozen diktes van de klei- + filterlaag. Ook hier blijkt dus het faalmechanisme “lokale afschuiving” een ondergeschikte rol te spelen.

[4] De berekeningen met ongunstige steenzetting (5 en 6) leveren zoals verwacht hogere uitkomsten dan de berekeningen met normale steenzetting (2, 3 en 4).

[5] De benodigde diktes voor asfaltbekleding zijn zoals verwacht lager dan voor steenbekleding (c.q blokken).

[6] Voor waterbouwasfaltbeton zijn de rekenresultaten onafhankelijk van het type filter. De rekenresultaten van berekening 7 en 8 zijn dan ook gelijk en hetzelfde geldt voor berekening 11 en 12.

[7] De benodigde dikte bij open steenasfalt is groter dan voor waterbouwasfaltbeton. De verschillen bedragen orde 1 dm voor filtertype geotextiel (vergelijk berekening 7 met berekening 9 en berekening 11 met berekening 13) en orde 2 cm voor het filtertype zandasfalt (vergelijk berekening 8 met berekening 10 en berekening 12 met berekening 14). De verklaring hiervoor zit in het feit dat in vergelijking (1) (zie hoofdstuk 2) het maximum doorgaans gelijk is aan  $C_3+C_4H_s$ , waarbij  $H_s$  gelijk is aan de golfhoogte en  $C_3$  en  $C_4$  twee constanten zijn. De waarden van constanten  $C_3$  en  $C_4$  voor de verschillende typen bekledingen en filters staan aangegeven in Tabel 5.4. Voor een golfhoogte van pakweg 2 m. geven deze parameters precies de verschillen die uit de berekening van Tabel 5.5 volgen. Uitzonderingen op de genoemde verschillen worden aangetroffen bij locatie Petten voor hogere herhalings tijden. Oorzaak hiervan is het gegeven dat bij locatie Petten in vergelijking (1) voor een aantal stormgebeurtenissen (en niet toevallig de meest extreme) het maximum gelijk is aan  $D_\alpha$  en de constanten  $C_3$  en  $C_4$  derhalve niet meer volledig bepalend zijn voor de verschillen in uitkomsten.

Tabel 5.4 Constanten  $C_3$  en  $C_4$  voor verschillende bekledingstypen

type bekleding+filter	$C_3$	$C_4$
waterbouwasfaltbeton + geotextiel	0.1	0.0167
waterbouwasfaltbeton + zandasfalt	0.1	0.0167
open steenasfalt + geotextiel	0.2	0.025
open steenasfalt + zandasfalt	0.1	0.025

[8] De uitkomsten voor ondergrond zand enerzijds en ondergrond klei anderzijds zijn exact gelijk, met uitzondering van een aantal berekeningen voor hoge herhalings tijden bij locatie Petten (vergelijk berekening 7, 8, 9 en 10 met resp. berekening 11, 12, 13 en 14). De verklaring hiervoor zit in het feit dat in vergelijking (1) het maximum doorgaans gelijk is aan  $C_3+C_4H_s$ . Het type ondergrond is niet van invloed op de waarde van waarden  $C_3$  en  $C_4$ . Slechts voor de hogere herhalings tijden bij de locatie Petten is in vergelijking (1) voor een aantal stormgebeurtenissen het maximum gelijk is aan  $D_\alpha$ , en op deze waarde is het type ondergrond wel van invloed.

Tabel 5.5 Benodigde laagdikte voor 3 locaties, 14 type bekledingen (zie Tabel 5.3) en 8 herhalingsstijden (T).

<b> Vlissingen </b>														
<b>T (jaar)</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>	<b>12</b>	<b>13</b>	<b>14</b>
<b>10</b>	0.48	0.48	0.48	0.48	0.64	0.64	0.12	0.12	0.24	0.14	0.12	0.12	0.24	0.14
<b>30</b>	0.51	0.52	0.52	0.52	0.70	0.70	0.13	0.13	0.24	0.14	0.13	0.13	0.24	0.14
<b>100</b>	0.55	0.57	0.57	0.57	0.75	0.75	0.13	0.13	0.24	0.14	0.13	0.13	0.24	0.14
<b>300</b>	0.59	0.61	0.61	0.61	0.81	0.81	0.13	0.13	0.25	0.15	0.13	0.13	0.25	0.15
<b>1000</b>	0.64	0.65	0.65	0.65	0.86	0.86	0.13	0.13	0.25	0.15	0.13	0.13	0.25	0.15
<b>2000</b>	0.66	0.67	0.67	0.67	0.89	0.89	0.13	0.13	0.25	0.15	0.13	0.13	0.25	0.15
<b>4000</b>	0.68	0.69	0.69	0.69	0.91	0.91	0.13	0.13	0.25	0.15	0.13	0.13	0.25	0.15
<b>10000</b>	0.71	0.71	0.71	0.71	0.95	0.95	0.14	0.14	0.25	0.15	0.14	0.14	0.25	0.15
<b> Den Helder </b>														
<b>T (jaar)</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>	<b>12</b>	<b>13</b>	<b>14</b>
<b>10</b>	0.39	0.37	0.37	0.37	0.49	0.49	0.13	0.13	0.24	0.14	0.13	0.13	0.24	0.14
<b>30</b>	0.42	0.40	0.40	0.40	0.53	0.53	0.13	0.13	0.24	0.14	0.13	0.13	0.24	0.14
<b>100</b>	0.44	0.43	0.43	0.43	0.57	0.57	0.13	0.13	0.25	0.15	0.13	0.13	0.25	0.15
<b>300</b>	0.46	0.46	0.46	0.46	0.61	0.61	0.13	0.13	0.25	0.15	0.13	0.13	0.25	0.15
<b>1000</b>	0.48	0.48	0.48	0.48	0.64	0.64	0.13	0.13	0.25	0.15	0.13	0.13	0.25	0.15
<b>2000</b>	0.50	0.50	0.50	0.50	0.66	0.66	0.14	0.14	0.25	0.15	0.14	0.14	0.25	0.15
<b>4000</b>	0.51	0.52	0.52	0.52	0.69	0.69	0.14	0.14	0.25	0.15	0.14	0.14	0.25	0.15
<b>10000</b>	0.53	0.54	0.54	0.54	0.71	0.71	0.14	0.14	0.26	0.16	0.14	0.14	0.26	0.16
<b> Petten </b>														
<b>T (jaar)</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>	<b>12</b>	<b>13</b>	<b>14</b>
<b>10</b>	1.00	1.04	1.04	1.04	1.38	1.38	0.14	0.14	0.26	0.16	0.14	0.14	0.26	0.16
<b>30</b>	1.14	1.19	1.19	1.19	1.56	1.56	0.14	0.14	0.26	0.16	0.14	0.14	0.26	0.16
<b>100</b>	1.30	1.34	1.34	1.34	1.75	1.75	0.14	0.14	0.27	0.17	0.14	0.14	0.27	0.17
<b>300</b>	1.43	1.47	1.47	1.47	1.90	1.90	0.15	0.15	0.27	0.19	0.15	0.15	0.27	0.17
<b>1000</b>	1.58	1.61	1.61	1.61	2.07	2.07	0.16	0.16	0.28	0.23	0.15	0.15	0.28	0.18
<b>2000</b>	1.67	1.69	1.69	1.69	2.16	2.16	0.18	0.18	0.28	0.26	0.15	0.15	0.28	0.18
<b>4000</b>	1.74	1.76	1.76	1.76	2.25	2.25	0.19	0.19	0.29	0.29	0.15	0.15	0.28	0.18
<b>10000</b>	1.85	1.85	1.85	1.85	2.36	2.36	0.21	0.21	0.32	0.32	0.16	0.16	0.29	0.19

### 5.3 Uniforme bekleding, optie ongunstigste belasting

In de tweede serie testberekeningen is wederom uitgegaan van een uniform bekledingstype. Dit keer echter zijn de berekeningen uitgevoerd met de optie “ongunstigste belasting”, d.w.z. de maximale belasting gedurende een stormgebeurtenis wordt geselecteerd als representatieve belasting. Vanwege de langere rekentijd bij deze optie is het aantal uitgevoerde berekeningen beperkt in vergelijking met de vorige paragraaf.

Voor een herhalingsijd van 4000 jaar zijn ontwerpberekeningen voor de volgende bekledingstypen uitgevoerd:

Tabel 5.6 Bekledingstypen waarvoor berekeningen zijn uitgevoerd met de optie “ongunstigste belasting”.

15.	Blokken
16.	Normale steenzetting met klei+filterlaag van 1 m
17.	Ongunstige steenzetting met klei+filterlaag van 1 m
18.	Waterbouwasfaltbeton met filtertype geotextiel
19.	Grasbekleding

Voor de laatste optie bleken de resultaten telkens nogal triviaal: voor alle herhalingsijden blijken de dijken niet te voldoen, zelfs niet in het geval de grasmat van goede kwaliteit is. De relatief lange tijdsduur van de opgeschaalde gebeurtenissen geven teveel belasting voor een dijk die van top tot teen met gras bedekt is. Opgemerkt wordt dat deze berekening nogal academisch van aard is, omdat een dijk die van top tot teen met gras bedekt in de praktijk niet voorkomt.

Voor de overige bekledingstypen zijn wel gewoon uitkomsten verkregen. Deze staan vermeld in Tabel 5.7 samen met de corresponderende berekening uit de vorige paragraaf. Met “corresponderende berekening” wordt bedoeld dat de bekledingstypen hetzelfde zijn, maar dat het verschil bestaat uit de keuze voor de optie “ongunstigste belasting” versus “hoogste waterstand”. Uit de tabel blijkt dat de verschillen minimaal zijn, waaruit maar weer eens blijkt dat de optie “hoogste waterstand” in veel situaties een goede benadering is voor de statistisch/fysisch meer correcte optie “ongunstigste belasting”.

Tabel 5.7 Resultaten van de omschreven berekeningen voor uniforme bekledingstypen.

Berekening:	15	1	verschil	16	3	verschil	17	4	verschil	18	6	verschil
<b>Vlissingen</b>	0.68	0.68	0.00	0.69	0.69	0.00	0.92	0.91	0.01	0.14	0.13	0.01
<b>Den Helder</b>	0.52	0.51	0.01	0.52	0.52	0.00	0.69	0.69	0.00	0.14	0.14	0.00
<b>Petten</b>	1.76	1.74	0.02	1.78	1.76	0.02	2.26	2.25	0.01	0.19	0.19	0.00

## 5.4 Frequentieberekeningen

In de vorige paragrafen zijn louter *ontwerpberekeningen* uitgevoerd voor de verschillende bekledingstypen, d.w.z. voor gegeven herhalingstijden/faalfrequenties is bepaald wat de benodigde laagdikte van de bekleding is. HYDRA-K biedt ook de mogelijkheid om middels *faalfrequentie-berekeningen* de omgekeerde weg te bewandelen, d.w.z. voor een gegeven combinatie van profiel, bekledingstype en dikte van de bekleding kan de faalfrequentie bepaald worden.

De resultaten van beide type berekeningen dienen uiteraard consistent te zijn. Testberekeningen tonen aan dat dit inderdaad het geval is, zij het dat door afrondingsfouten niet exact de gewenste uitkomst berekend wordt. Dit wordt toegelicht aan de hand van een rekenvoorbeeld:

**Voorbeeld.** In paragraaf 5.2 is in de derde berekening (d.w.z. normale steenzetting met een klei+filterlaag van 1 m) voor locatie Petten bij een herhalingstijd van 4.000 jaar berekend dat de benodigde laagdikte van de bekleding gelijk is aan 1.76 m. Als deze dikte vervolgens aangenomen wordt in een faalfrequentie-berekening geeft HYDRA-K een herhalingstijd van 3887 op als resultaat, ofwel 123 jaar *lager* dan de beoogde 4.000 jaar. Echter, indien een laagdikte van 1.77 m. (1 cm hoger) wordt ingevoerd dan resulteert een herhalingstijd van 4507 jaar, ofwel ongeveer 500 jaar *hoger* dan de beoogde 4000 jaar.

Een verandering in de laagdikte van 1 cm kan dus honderden jaren verschil opleveren in de herhalingstijd. Het is daarom niet reëel te verwachten dat HYDRA-K in het bovenstaande voorbeeld *exact* uitkomt op de oorspronkelijk ingevoerde herhalingstijd van 4000 jaar. Omgekeerd betekent de grote gevoeligheid van de herhalingstijd voor een verandering in de laagdikte dat de laagdikte juist relatief ongevoelig is voor een verandering in de herhalingstijd. Met andere woorden: als de controleberekening op de laagdikte wordt uitgevoerd in plaats van op de herhalingstijd dan wordt de beoogde laagdikte wel exact op de centimeter nauwkeurig berekend.

De belangrijkste conclusie uit de testberekeningen is in elk geval dat de consistentie van de beide type berekeningen (faalfrequentie-berekening en ontwerpberekening) afdoende gewaarborgd is.

## 5.5 Berekeningen voor delen van het profiel

In de vierde serie testberekeningen worden delen van het profiel doorgerekend, waarmee impliciet is aangenomen dat het dijkprofiel geen uniform bekledingstype heeft. Omdat bij berekeningen voor delen van het dijkprofiel het verloop van de storm van belang is zijn de berekeningen uitgevoerd met de optie “ongunstigste belasting”, d.w.z. de maximale belasting gedurende een stormgebeurtenis wordt geselecteerd als representatieve belasting.



## Gras, hoog op het talud

In één van de vorige testberekeningen is aangetoond dat een dijkprofiel dat volledig met gras bedekt is, tot onacceptabel grote faalfrequenties leidt (zie paragraaf 5.3). In genoemde paragraaf werd daarbij opgemerkt dat de berekening nogal academisch van aard is, omdat een dijk die van top tot teen met gras bedekt in de praktijk niet voorkomt. In de praktijk wordt grasbekleding alleen aangetroffen op de hogere delen van het talud. Om de berekening voor gras van het “academische karakter” te ontdoen is daarom een berekening uitgevoerd voor een dijkprofiel waarbij zich alleen op het hogere deel van het talud gras bevindt.

In eerste instantie is voor de drie locaties een ontwerpberekening uitgevoerd bij een herhalingstijd van 4.000 jaar voor onderstaand dijkdeel.

20. grasbekleding van 7 m + NAP tot aan de kruin
--

Voor twee van de drie locaties (Den Helder en Petten) blijkt uit de ontwerpberekening dat wederom geen enkele kwaliteit grasmat te voldoen aan de vereiste faalfrequentie van 1/4.000 jaar. Voor Vlissingen daarentegen wordt wel voldaan aan de faalfrequentie (1/4.000 jaar) indien de grasmat van goede kwaliteit is. Hieruit blijkt dat het voor gras wel degelijk mogelijk is om aan de toetscriteria te voldoen.

Om het verloop in benodigde kwaliteit grasmat te testen is locatie Vlissingen voor meerdere herhalingstijden doorgerekend. Tabel 5.8 geeft de resultaten weer en daarin is te zien dat zoals verwacht de eisen aan de kwaliteit van de grasmat minder streng zijn naarmate de herhalingstijd afneemt. Voor lagere herhalingstijden (10 en 30 jaar) bleek overigens dat de grasbekleding niet belast werd, d.w.z. geen van de opgeschaalde stormen kwam tot een golfplooptniveau van boven de 7 m +NAP.

Tabel 5.8 ontwerpberekeningen voor gras, locatie Vlissingen

herhalingstijd	ontwerp kwaliteit
10	slecht
30	slecht
100	slecht
300	slecht
1000	matig
2000	matig
4000	goed
10000	goed

## Steenzetting, laag op het talud

Bij bekledingen die laag op het talud gelegen zijn speelt het aspect dat deze bij extreme stormgebeurtenissen voornamelijk belast wordt in “de flanken” van de storm (d.w.z. de aanloop en afloop), terwijl tijdens de hoogste waterstanden er nauwelijks sprake is van belasting omdat het dan onder water gelegen is. Bij het opschalen van de stormgebeurtenissen is het is maar zeer de vraag of de modellering van de aanloop en uitloop van de storm nog in orde is.

Als testberekening is een dijkdeel met steenbekleding doorgerekend bij locatie Vlissingen (zie onder voor details). Tabel 5.9 bevat de resultaten van deze exercitie.

21. Normale steenzetting met klei+filterlaag van 1 m., bereik tussen 0 m +NAP en 2.5 m +NAP
---

Tabel 5.9 ontwerpberoeeningen voor steenbekleding locatie Vlissingen

herhalingstijd	ontwerpdikte
10	0.48
30	0.51
100	0.52
300	0.44
1000	geen uitkomst
2000	geen uitkomst
4000	geen uitkomst
10000	geen uitkomst

Uit de tabel blijkt dat voor herhalingstijden tot 300 jaar een ontwerpdikte voor steenbekleding wordt berekend. Voor grotere herhalingstijden, echter, geeft HYDRA-K geen uitkomst. Uit nadere analyse blijkt dat voor deze hoge herhalingstijden de waterstanden dusdanig ver opgeschaald wordt dat het relatief laag gelegen dijkdeel in het geheel niet belast wordt gedurende alle doorgerekende stormen. In formule: de waarde van  $z_t$  (zie stap 1 paragraaf 2.4) is na opschaling gedurende elke tijdstap van elke storm hoger dan het niveau van de bovenkant van de bekleding.

Het bovenstaande impliceert dat alle simultane waarnemingen uit het faaldomein geschoven zijn, hetgeen duidt op een zogenaamd “niet-regulier faaldomein”. In hoofdstuk **Error! Reference source not found.** wordt nader op dit fenomeen ingegaan.

## 6 Niet-reguliere faaldomeinen

### 6.1 Probleembeschrijving

#### 6.1.1 Illustratie aan de hand van een rekenvoorbeeld

De laatste testberekening van het vorige hoofdstuk (Tabel 5.9) is een voorbeeld van een situatie waar het concept van HYDRA-K niet naar behoren functioneert. Dit is het best te illustreren aan de hand van Figuur 6.1 t/m Figuur 6.3, waarin voor een aantal windrichtingssectoren het faaldomein is weergegeven in het standaard exponentiële vlak voor herhalingsstijden van 30, 100 en 300 jaar. Behalve het faaldomein zijn ook de werkelijke en de opgeschaalde simultane waarnemingen weergegeven, waarbij de kleur van de waarnemingen de heersende windrichting aanduidt.

De lengte van de opschaling in de figuren is afhankelijk van de herhalingsstijd. Dit laatste behoeft wellicht enige toelichting. De lengte van de verschuiving in het standaard exponentiële vlak wordt binnen HYDRA-K (bij een ontwerpberekening) als volgt bepaald:

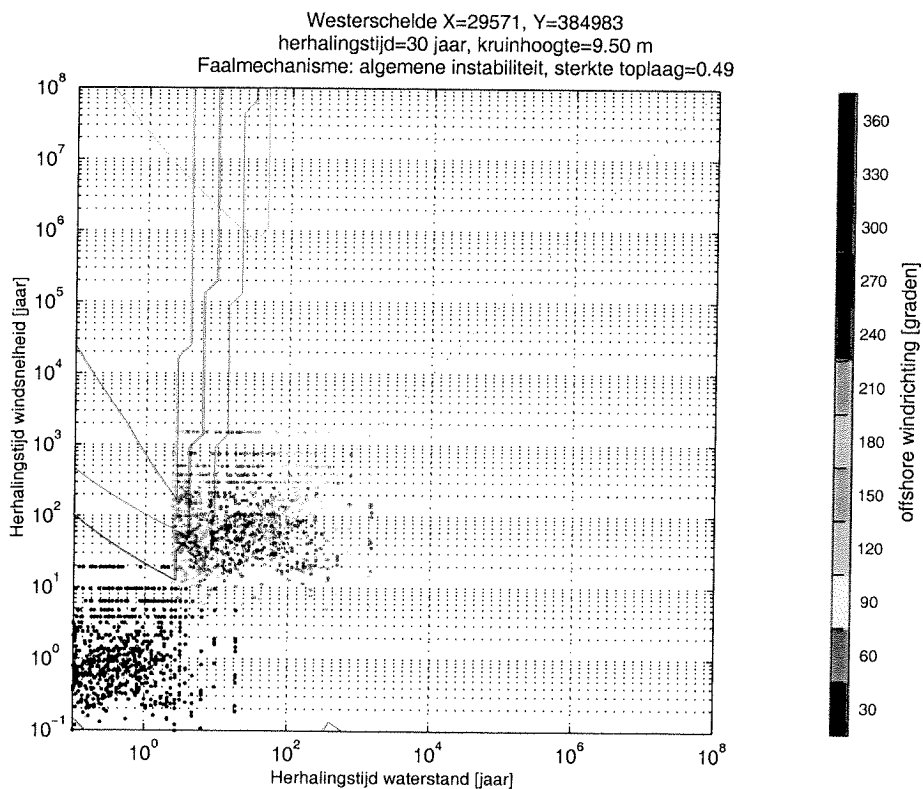
$$\lambda = -\ln\left(\frac{v\Delta}{\kappa}\right) = \ln(\kappa T) - \ln(\Delta) \quad (21)$$

Waarin:

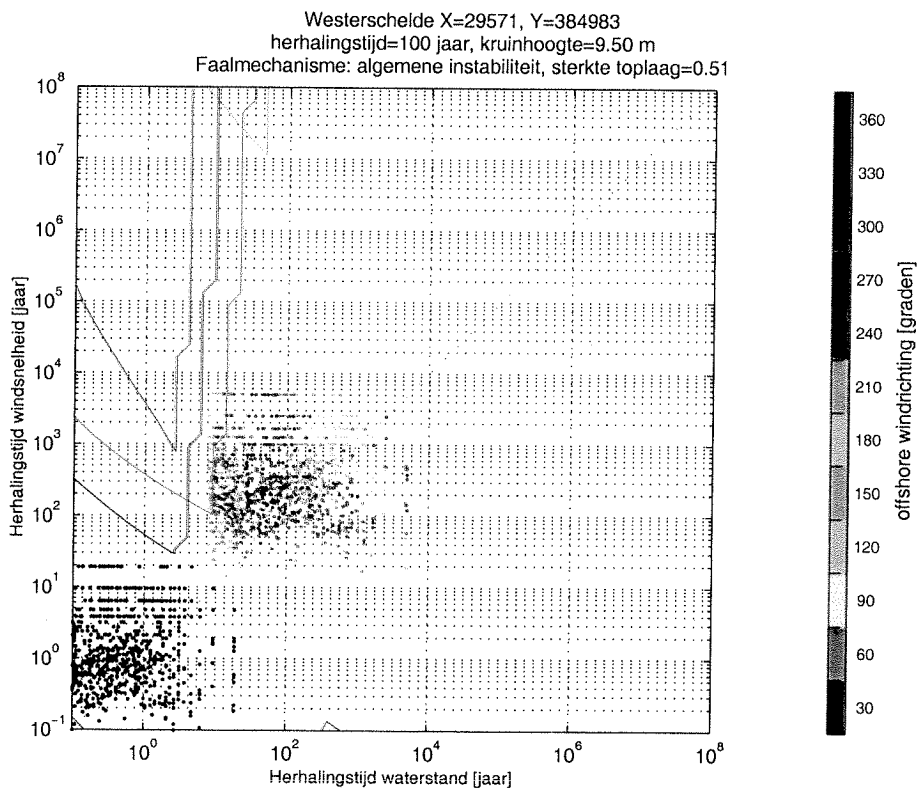
- $\lambda$  = de verschuiving in het standaard exponentiële vlak.
- $v$  = de gewenste jaarlijkse faalfrequentie.
- $T$  = de gewenste herhalingsstijd van falen (ofwel de inverse van  $v$ ).
- $\kappa$  = het aantal punten dat in het faalgebied verschoven is.
- $\Delta$  = de lengte van de periode waarin de waarnemingen verzameld zijn.

Bij uitvoering van een ontwerpberekening is het rechterlid van bovenstaande vergelijking bekend ( $\kappa$  en  $T$  dienen door de gebruiker opgegeven te worden en  $\Delta$  is het aantal jaren van de meetreeks). De waarde van  $\lambda$  kan derhalve direct bepaald worden, zodat de grootte van de verschuiving van de waargenomen data in het standaard exponentiële vlak bekend is.

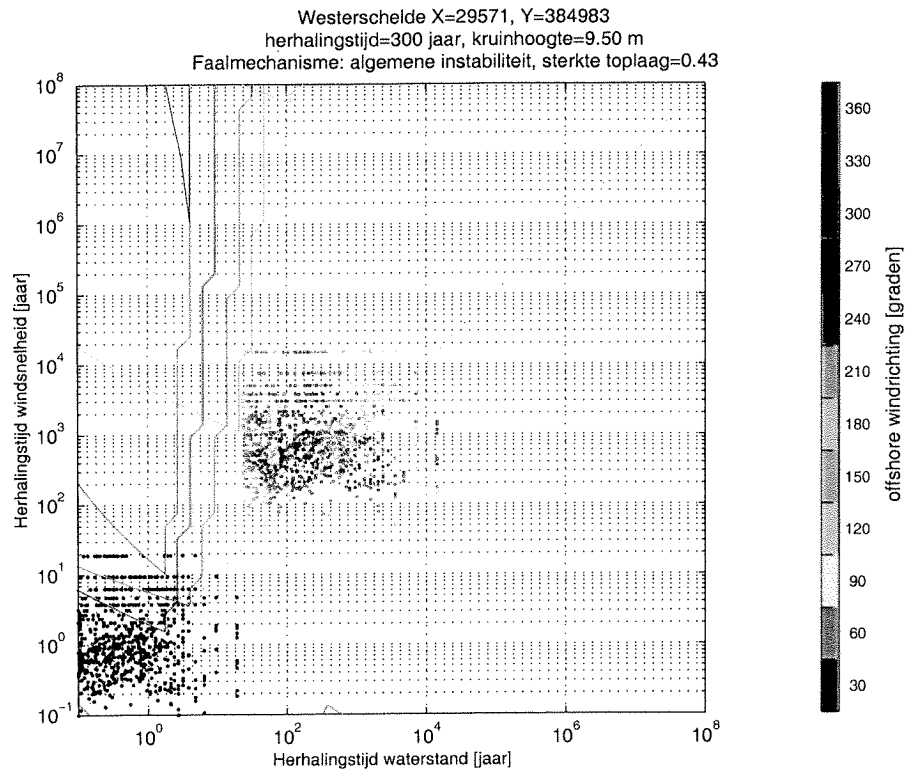
Uit deze vergelijking blijkt dat de lengte van de verschuiving in het standaard exponentiële vlak toeneemt naarmate de gewenste herhalingsstijd van falen toeneemt. Uit Figuur 6.1 t/m Figuur 6.3 blijkt dat als de verschuiving maar groot genoeg is, alle waarnemingen uit het faaldomein verdwijnen. Tabel 5.9 geeft aan dat dit het geval is voor een herhalingsstijd van 4.000 jaar (of groter). Helaas kan dit niet in een figuur gedemonstreerd worden, omdat het faaldomein voor deze herhalingsstijd niet vastgelegd kan worden (immers de dikte van de bekleding is niet bekend omdat de ontwerpberekening geen uitkomst geeft). Vandaar dat in Figuur 6.4 wel de verschoven simultane waarnemingen getoond kunnen worden maar niet het faaldomein.



Figuur 6.1 Verschuiven van de simultane waarnemingen ten opzichte van het (windrichtingafhankelijke) faaldomein. De verschuiving is die voor een herhalingstijd van 30 jaar.

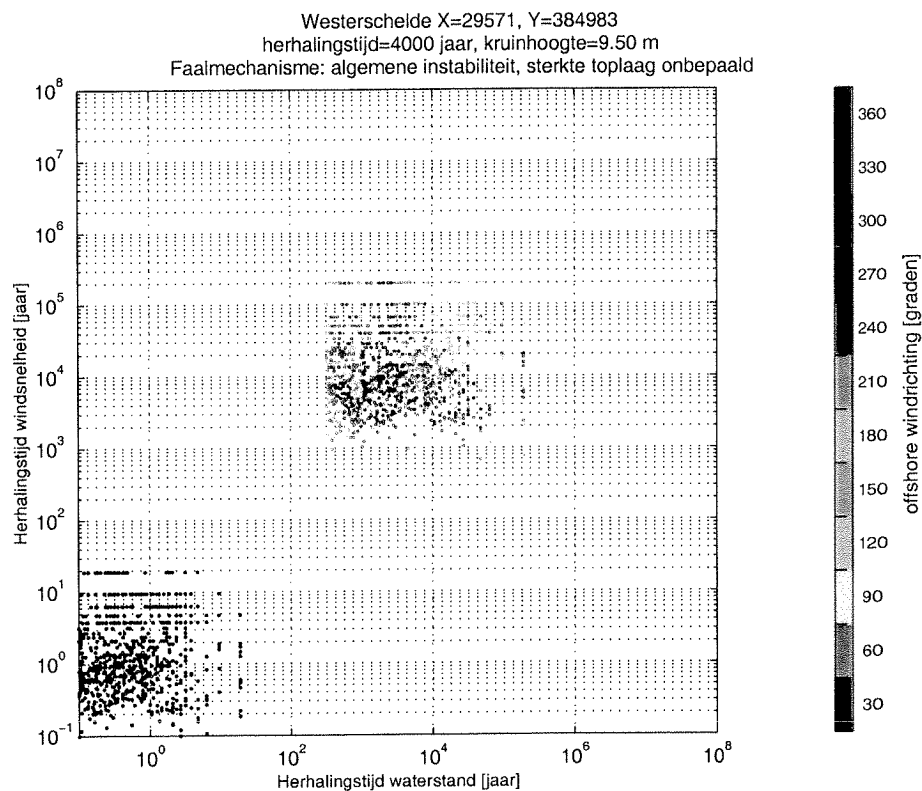


Figuur 6.2 Verschuiven van de simultane waarnemingen ten opzichte van het (windrichtingafhankelijke) faaldomein. De verschuiving is die voor een herhalingstijd van 100 jaar.



18-Nov-2003 18:48:20

Figuur 6.3 Verschuiven van de simultane waarnemingen ten opzichte van het (windrichtingafhankelijke) faaldomein. De verschuiving is die voor een herhalingstijd van 300 jaar.

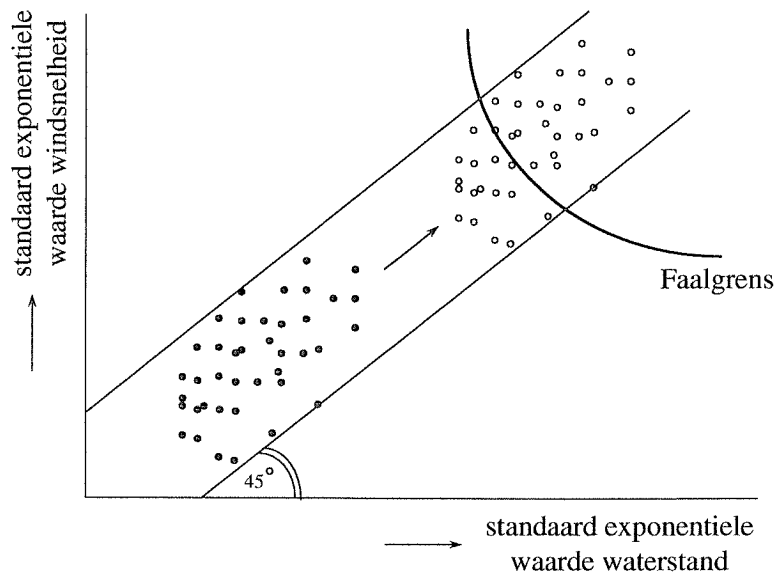


18-Nov-2003 19:34:01

Figuur 6.4 Verschuiven van de simultane waarnemingen ten opzichte van het (windrichtingafhankelijke) faaldomein. De verschuiving is die voor een herhalingstijd van 4000 jaar.

## 6.1.2 Probleembeschrijving algemeen

Het concept van HYDRA-K gaat uit van “reguliere faaldomeinen” (zie Figuur 6.5), d.w.z. het principe dat bij toename van de verschuiving de simultane waarnemingen vanzelf in het faaldomein terecht komen en er niet meer uitgaan bij verdere toename van de verschuiving. Dergelijke faaldomeinen ontstaan wanneer de hydraulische belasting toeneemt naarmate de windsnelheid en de waterstand toenemen, zoals bijvoorbeeld het geval is bij de faalmechanismen “golfoploop” en “golfoverslag”.



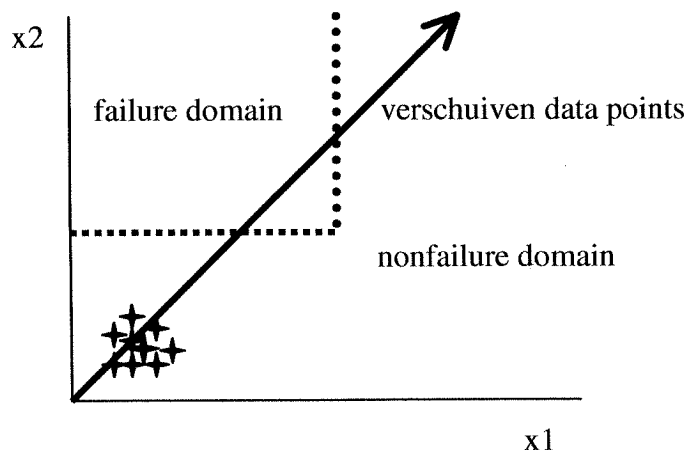
Figuur 6.5 Schematische weergave van de verschuiving van waarnemingen in het standaard exponentiële vlak

Echter, voor de faalmechanismen die in dit project aan HYDRA-K zijn toegevoegd is de hydraulische belasting niet meer per definitie een strikt stijgende functie van de waterstand en windrichting, vooral niet als slechts een deel van het dijkprofiel getoetst wordt. Immers, bij een toename in de waterstand kan de stilwaterlijn op een gegeven moment zó ver boven het getoetste dijkdeel liggen dat er van belasting geen sprake meer is.

## 6.2 Mogelijke oplossingsrichtingen

### 6.2.1 Andere verschuiving in het standaard exponentiële vlak

In het rekenvoorbeeld van paragraaf 5.5 en paragraaf 6.1.1 was de verschuiving van de waarnemingen in het standaard exponentiële vlak soms zó groot dat laaggelegen delen van het dijkprofiel tijdens geen van de opgeschaalde stormgebeurtenissen werd belast. De simultane waarnemingen zijn dan feitelijk uit het faaldomein geschoven, op de wijze zoals in Figuur 6.6 schematisch is weergegeven voor een fictief faaldomein.



Figuur 6.6 Schets van een niet-regulier faaldomein

Een mogelijke oplossingsrichting is om de waarde van de verschuiving anders te kiezen dan nu gedaan wordt in HYDRA-K, d.w.z. niet meer volgens vergelijking (21). In plaats daarvan wordt  $\lambda$  op iteratieve wijze gevarieerd tot het moment dat voldoende waarnemingen in het faaldomein liggen. Vervolgens wordt de faalfrequentie bepaald uit de volgende vergelijking:

$$v = \frac{1}{T} = \frac{e^{-\lambda} \kappa}{\Delta} \quad (22)$$

Waarin:

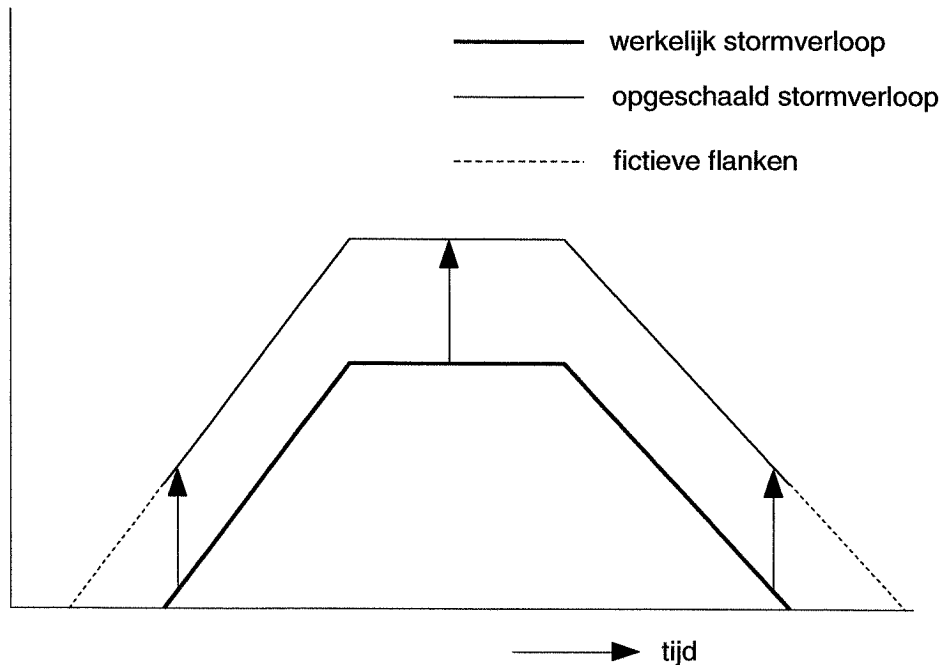
- $\lambda$  = de uitgevoerde verschuiving in het exponentiële vlak.
- $v$  = de jaarlijkse faalfrequentie.
- $T$  = de gewenste herhalingstijd van falen (ofwel de inverse van  $v$ ).
- $\kappa$  = het aantal punten dat in het faalgebied verschoven is.
- $\Delta$  = de lengte van de periode waarin de waarnemingen verzameld zijn.

Vergelijking (22) is in feite niets anders dan een herschreven versie van vergelijking (21). Het belangrijkste verschil is dat de waarde van  $\kappa$  nu niet door de gebruiker opgegeven wordt, maar de uitkomst is van tellen na uitvoering van de iteratieve verschuivingprocedure.

In fase 3 van dit project wordt aan de hand van een gestileerd probleem geanalyseerd of met deze methode het probleem van de niet-reguliere faaldomeinen (gedeeltelijk) wordt opgelost.

## 6.2.2 Kunstmatige verlenging van stormgebeurtenissen

In Figuur 6.7 is een opgeschaald stormverloop schematisch weergegeven (het is in dit verband even onbelangrijk of hier de waterstand, de windsnelheid op de windopzet is weergegeven). Een probleem bij het opschalen is dat de stormgebeurtenis met relatief hoge waterstanden en windsnelheden aanvangt en dus geen “normale” aanloop heeft. Het kan daardoor gebeuren dat de lage delen op het dijkprofiel in het geheel niet belast worden tijdens deze opgeschaalde stormgebeurtenis, hetgeen de belangrijkste oorzaak is voor de eerder besproken niet-reguliere faaldomeinen.



Figuur 6.7 Schematische weergave van een opgeschaald stormverloop met fictieve aanloop en uitloop.

Een mogelijkheid om dit probleem tegen te gaan is om het stormverloop kunstmatig te verlengen (zie Figuur 6.7). Op deze wijze wordt gerealiseerd dat ook de lagere delen van het dijkprofiel belast worden en wordt mogelijk het fenomeen van niet-reguliere faaldomeinen voorkomen.

Indien voor deze oplossing gekozen wordt, moeten eerst de volgende vraagstukken beantwoord worden:

- Op welke wijze dienen de lijnen van het opgeschaalde stormverloop te worden doorgetrokken?
- Hoe is de relatie tussen windsnelheid en waterstand in de aanloop (en uitloop)?

Beiden dienen beantwoord te worden op basis van de fysica c.q. metingen van wind en waterstand. In de vervolgfases van dit project wordt verder op deze vraagstukken ingegaan en wordt geanalyseerd of met deze methode het fenomeen van niet-reguliere faaldomeinen voorkomen kan worden.



## 7 Conclusies

Uitgaande van versie 2.05 van HYDRA-K is een demonstratiemodel gemaakt waaraan, in vergelijking met HYDRA-K, de bekledingstypen gras, asfalt en steenzetting zijn toegevoegd. Daarbij zijn de volgende faalmechanismen geïmplementeerd:

- *asfalt*: schade aan de bekleding als gevolg van golfklappen.
- *steenzetting*: instabiliteit van de top laag en materiaal transport als gevolg van golven.
- *gras*: schade aan de bekleding als gevolg van golfklappen en als gevolg van stroming op het buitentalud.

De user-interface van HYDRA-K is aangepast zodat op gebruikersvriendelijke wijze dijkprofielen gedefinieerd kunnen worden met meerdere typen bekledingen. Ook voor dergelijke samengestelde dijkprofielen kan nu met HYDRA-K de faalkans berekend worden. Echter, de leidraad toetsen (TAW, 1999) doet geen uitspraken over faalkansen van dijken die uit meerdere bekledingen bestaan. Volgens de leidraad moeten de afzonderlijke delen ook afzonderlijk getoetst worden en zolang elk van deze delen aan de gestelde eisen voldoet is voor de dijk als geheel voldoende. Om de berekeningen met HYDRA-K zoveel mogelijk te laten stroken met de leidraad, wordt daarom aangeraden om per berekening slechts één bekledingstype door te rekenen.

Na voltooiing van de implementatie zijn testberekeningen uitgevoerd voor de nieuwe faalmechanismen en bekledingstypen. Voor drie locaties (Vlissingen, Den Helder en Petten) zijn in eerste instantie ontwerpberekeningen uitgevoerd voor dijkprofielen met uniforme bekledingstypen. Geen van de resultaten bleek in strijd met datgene wat op voorhand verwacht kon worden. Het model lijkt derhalve betrouwbare uitkomsten te leveren.

Vervolgens zijn ontwerpberekeningen uitgevoerd voor delen van dijkprofielen en toen bleek dat het model in een aantal situaties geen uitkomst geeft. Uit nadere analyse kwam naar voren dat de waterstanden in HYDRA-K soms zó ver worden opgeschaald dat relatief laag gelegen dijkdelen in het geheel niet belast worden. Immers, bij een toename in de waterstand kan de stilwaterlijn op een gegeven moment zó ver boven het getoetste dijkdeel liggen dat er van belasting geen sprake meer is. Bij een grafische weergave van verschoven simultane waarnemingen van wind en waterstand resulteert dit effect in een zogenaamd “niet-regulier faaldomein”. De hydraulische belasting is in dit geval niet meer een strikt stijgende functie van de waterstand en windrichting, zodat de simultane waarnemingen bij een toename in de grootte van de verschuiving uit het faaldomein kunnen verschuiven.

In dit rapport zijn twee oplossingsrichtingen aangedragen om te realiseren dat ook voor niet-reguliere faaldomeinen faalfrequentie-berekeningen en ontwerpberekeningen uitgevoerd kunnen worden. Het eerste betreft een andere keuze voor de grootte van de verschuiving van de simultane waarnemingen, zodat er voldoende waarnemingen in het faaldomein verschoven worden. Het tweede betreft een methode om op synthetische wijze de opgeschaalde stormgebeurtenissen te verlengen, zodat ook de lager gelegen delen tijdens elke storm belast worden. Beide methoden worden verder uitgewerkt en geanalyseerd in de vervolgfases van dit project.

## 8 Referenties

HKV, 2000: HYDRA-K, functioneel ontwerp, HKV LIJN IN WATER; december 2000.

TAW, 1999: Leidraad toetsen op veiligheid, Technische adviescommissie voor de waterkeringen (TAW), augustus 1999.

TNO bouw, 2003: *Theoriehandleiding PC-ring. Deel A: Mechanismebesrijving*, Steenbergen, H.M.G.M. en Vrouwenvelder, A.C.W.M., TNO-rapport 2003-CI-R0020. april 2003.

WL, 2003: Geavanceerde toetsing(a,b) bij de Houtribsluizen, project H4169, juni, 2003.

VTV'03, Voorschrift toetsen op Veiligheid, verschijnt eind 2003

Steenstoets 3.20, documentatie wij steen toets 3.20, WL, maart 2000



# **BIJLAGEN**

## A Aanpassingen in de code

In deze bijlage wordt de code van de nieuwe routines getoond en van de belangrijkste code wijzigingen:

### Stat\StatBerekenZ

```
function Z = StatBerekenZ (FaalMechanisme, Regio, Locatie, correctieRVB, ...
    Profiel, WaarnOverschrFactor, WaarnStormSectorIndex, Waarneming, WaarnStormIndex, ...
    Stat, StroomCor, faalMechID)

%STATBEREKENZ      In StatBerekenZ wordt de Z-functie uitgerekend voor een gegeven
%                  faalmechanisme. Deze functie is beschreven in 3.1.1 van het
%                  functioneel ontwerp
%                  Merk op, dat de case "waterstand" bedoeld is om de
%                  controlefunctie te kunnen bepalen in verband met
%                  het Randvoorwaardenboek.
% AANROEP
%   Z = StatBerekenZ (FaalMechanisme, Locatie,correctieRVB, Profiel, ...
%   WaarnOverschrFactor, WaarnStormSectorIndex,Waarneming,WaarnStormIndex,Stat)
% INVOER
%   FaalMechanisme:      het faalmechanisme ('golfoploop', 'golfoverslag' of
%   'blokkeninstabiliteit')
%   Locatie:             de x,y coördinaten van de locatie
%   correctieRVB:        de correctie op de waterstand voor verschillen met het
Randvoorwaardenboek
%   Profiel:             het dijkprofiel
%   WaarnOverschrFactor: formule: -log(waarnemingenoverschrijdingsaantallen)/kappa
%                       (waterstand, windsnelheid, piekperiode, sign. golfhoogte)
%   WaarnStormSectorIndex: de sectorindex van de storm (windrichting van de storm)
%   Waarneming:          de golfrichting en de windrichting van de waarnemingen
%   WaarnStormIndex:     de index van de storm behorend bij de waarneming
%   Stat:                de Weibull parameters voor de waterstand, windsnelheid,
%                       piekperiode en sign. golfhoogte
% UITVOER
%   Z:                  een vector met Z-waarden

% EXTERNE FUNCTIES
%   Offshore2Nearshore
%   StatBerekenZGolfoploop
%   StatBerekenZGolfoverslag
%   StatBerekenZBlokinstabiliteit

% Copyright © 2000 Rijkswaterstaat/RIKZ All Rights Reserved.

% Auteur: Abe Hoekstra/Ingrid Lammers, HKV lijn in water

% $Header: /PR/313/HYDRA_K/Stat/StatBerekenZ.m 3      3-11-00 10:37 Hoekstra $
% $NoKeywords: $

GlobalConstants

% Allereerst worden de offshore gegevens bepaald met de functie StatWeibullInv uit de
% overschrijdingsfrequenties van de waarnemingen
% Daarna worden de offshoregegevens getransformeerd naar nearshore gegevens met de functie
Offshore2Nearshore
% Vervolgens wordt afhankelijk van het faalmechanisme de betreffende Z functie uitgerekend

Var_of= [StatWeibullInv(WaarnOverschrFactor,WaarnStormSectorIndex,Stat,Regio) Waarneming];

% De offshoregegevens bestaan uit:
% 1: waterstand
% 2: windsnelheid
% 3: piekperiode
% 4: significante golfhoogte
% 5: golfrichting
% 6: windrichting
% 7: noodsluiting (Oosterschelde)

% transformatie van offshore naar nearshore
```

```

% De nearshoregegevens bestaan uit:
% 1: waterstand
% 2: windsnelheid
% 3: piekperiode
% 4: significante golfhoogte
% 5: golfrichting
% 6: windrichting

Var_ne = Offshore2Nearshore (
Regio, Locatie.x, Locatie.y, correctieRVB, Var_of, StroomCor, faalMechID);

for i = 1:length(Profiel)
    switch FaalMechanisme.naam

        case 'golfoploop',
            [alfa_repr, gamma_b, gamma_f, hoogte_2pr, s_op, zeta, z_2pr] ...
            = StatBerekenZGolfoploop (Profiel(i), Var_ne(:,4), Var_ne(:,3), Var_ne(:,1),
Var_ne(:,5));
            Z(:,i) = FaalMechanisme.golfoploop.Hc(i) - Var_ne(:,1)-z_2pr;

        case 'waterstand',
            Z(:,i) = FaalMechanisme.golfoploop.Hc(i) - Var_ne(:,1);

        case 'golfoverslag',
            Z(:,i) = StatBerekenZGolfoverslag (Profiel(i), FaalMechanisme.golfoverslag.Qc, ...
            Var_ne(:,4), Var_ne(:,3), Var_ne(:,1), Var_ne(:,5));

        case 'blokkeninstabiliteit',
            if length(FaalMechanisme.blokkeninstabiliteit.Blok) == 1,
                Z(:,i) = StatBerekenZBlokinstabiliteit (Profiel(i), ...
                FaalMechanisme.blokkeninstabiliteit.Blok, ...
                Var_ne(:,4), Var_ne(:,3), Var_ne(:,1), Var_ne(:,5));
            else
                Z(:,i) = StatBerekenZBlokinstabiliteit (Profiel(i), ...
                FaalMechanisme.blokkeninstabiliteit.Blok(i), ...
                Var_ne(:,4), Var_ne(:,3), Var_ne(:,1), Var_ne(:,5));
            end

        case {'algemene instabiliteit', 'algemene instabiliteit met gras'}
            AantalLagenDijkbekleding = length(FaalMechanisme.dijkopbouw);
            ZWaardeLaagZi=zeros(size(Var_ne,1),AantalLagenDijkbekleding);
            for zi=1:AantalLagenDijkbekleding
                switch FaalMechanisme.dijkopbouw(zi).bekleding
                    case 'asfalt'
                        ZWaardeLaagZi(:,zi) = StatBerekenZAsfalt (Profiel(i), ...
                        FaalMechanisme.dijkopbouw(zi), ...
                        Var_ne(:,4), Var_ne(:,3), Var_ne(:,1), Var_ne(:,5));
                    case 'steenzetting'
                        ZWaardeLaagZi(:,zi) = StatBerekenZSteenzetting ...
                        (Profiel(i), FaalMechanisme.dijkopbouw(zi), ...
                        Var_ne(:,4), Var_ne(:,3), Var_ne(:,1), Var_ne(:,5));
                    case 'gras'
                        ZWaardeLaagZi(:,zi) = StatBerekenZGras (Profiel(i), ...
                        FaalMechanisme.dijkopbouw(zi), WaarnStormIndex, ...
                        Var_ne(:,4), Var_ne(:,3), Var_ne(:,1), Var_ne(:,5));
                    case 'blokken'
                        ZWaardeLaagZi(:,zi) = StatBerekenZBlokinstabiliteit ...
                        (Profiel(i), FaalMechanisme.dijkopbouw(zi), ...
                        Var_ne(:,4), Var_ne(:,3), Var_ne(:,1), Var_ne(:,5));
                end
            end
            %
            % Bepaal minimum Z-waarde over alle lagen ...
            %
            Z(:,i)=min(ZWaardeLaagZi,[],2);
        end
    end
    %
    % Bepaal minimum Z-waarde over alle profielen ...
    %
    Z = min(Z,[],2);

    Z = real(Z);
    nanind = find(isnan(Z));
    Z(nanind) = -Inf;

% De nearshoregegevens bestaan uit:

```

```
% 1: waterstand
% 2: windsnelheid
% 3: piekperiode
% 4: significante golfhoogte
% 5: golfrichting
% 6: windrichting
```

## Stat\StatBerekenZAsfalt

```
function [Z,q] = StatBerekenZAsfalt (Profiel, ZSegment, H_s, T_p, sl, theta_0)

%STATBEREKENZASFALT In StatBerekenZAsfalt wordt de Z-functie uitgerekend voor
% het faalmechanisme instabiliteit voor asfalt.
%
% AANROEP
% [Z,q] = BerekenZAsfalt (Profiel, ZSegment, H_s, T_p, sl, theta_0)
% INVOER
% Profiel: het dijkprofiel
% ZSegment: karakteristiek van het dijksegment (ondergrens & bovengrens,
% en bekledingsgegevens.
% H_s: significante golfhoogte (nearshore)
% T_p: de piekperiode (nearshore)
% sl: waterstand (nearshore)
% theta_0: golfrichting (nearshore) (t.o.v. het noorden, in radialen)
% UITVOER
% Z: Z is gelijk aan de asfaltsterkte minus de belasting
% q: de belasting q

% Auteur: Bert Jagers, WL | Delft Hydraulics

% 17-10-2003

laagdikte = ZSegment.bekledingsgegevens.laagdikte;
ondergrondtype = ZSegment.bekledingsgegevens.ondergrondtype; %'zand','klei'
asfalttype = ZSegment.bekledingsgegevens.asfalttype;
%'waterbouwasfaltbeton','opensteenafalt'
filtertype = ZSegment.bekledingsgegevens.filtertype; %'geotextiel','zandasfalt'
z_ondergrens = ZSegment.ondergrens;
z_bovengrens = ZSegment.bovengrens;

%helling = BepaalHelling(1.5,H_s,sl,Profiel);
helling = repmat(Profiel.helling_laag,size(sl));
bovenberm = sl>Profiel.hoogte_berm;
helling(bovenberm) = Profiel.helling_hoog;

sterkte = repmat(laagdikte,size(sl));
belasting = repmat(0,size(sl));

sltelaag = sl < (z_ondergrens - 0.25*H_s);
sltehoog = sl > (z_bovengrens + 2.0 *H_s);
resterend = ~sltelaag & ~sltehoog;

switch asfalttype
case 'waterbouwasfaltbeton'
%een_op=1./helling;
%if any(een_op<2)
% warning('Steile helling afgezwakt tot 1:2 t.b.v. berekening stabiliteit asfalt.')
% een_op(een_op<2)=2;
%end
switch ondergrondtype
case 'zand'
%een_op = min(floor(een_op),3);
%c1 = [-1 0.0248 0.0062];
%c2 = [-1 2.0306 2.2884];
minimum_laagdikte1 = (-0.18401+0.125056*H_s+0.0014*H_s.^2).*(-
1.69947+5.174361*helling+0.098738*H_s);
case 'klei'
%een_op = min(floor(een_op),6);
%c1 = [-1 0.0785 0.0348 0.0158 0.0101 0.0022];
%c2 = [-1 1.7497 1.8953 2.0742 2.0348 2.6552];
minimum_laagdikte1 = (-0.06927+0.084858*H_s+0.013069*H_s.^2).*(-
0.93087+5.108631*helling+0.060252*H_s);
end
%minimum_laagdikte1 = c1 * H_s.^c2;
minimum_laagdikte2 = 0.1 + 0.0167*H_s;
minimum_laagdikte = max(minimum_laagdikte1,minimum_laagdikte2);
belasting(resterend) = minimum_laagdikte(resterend);
```

```

tehogegolven      = resterend & (H_s>6.0);
sterkte(tehogegolven) = -inf;

case 'opensteenafalt'
%een_op=1./helling;
%if any(een_op<2)
% warning('Steile helling afgezwakt tot 1:2 t.b.v. berekening stabiliteit asfalt.')
% een_op(een_op<2)=2;
%end
switch ondergrondtype
case 'zand'
%een_op = min(floor(een_op),2);
%c1     = [-1 0.0341];
%c2     = [-1 2.3601];
minimum_laagdikte1 = (-0.05785+0.025545*H_s+0.026866*H_s.^2).*(-
0.93087+5.108631*helling+0.060252*H_s);
case 'klei'
%een_op = min(floor(een_op),4);
%c1     = [-1 0.1333 0.0464 0.0075];
%c2     = [-1 2.0074 2.2203 3.0382];
minimum_laagdikte1 = (-0.1229+0.17821*H_s+0.03541978*H_s.^2).*(-
1.36632+5.138124*helling+0.128026*H_s);
end
%minimum_laagdikte1 = c1 * H_s.^c2;
switch filtertype
case 'geotextiel'
c3     = 0.2;
c4     = 0.025;
case 'zandasfalt'
c3     = 0.1;
c4     = 0.025;
end
minimum_laagdikte2 = c3 + c4*H_s;
minimum_laagdikte = max(minimum_laagdikte1,minimum_laagdikte2);
belasting(resterend) = minimum_laagdikte(resterend);

tehogegolven      = resterend & (H_s>4.0);
sterkte(tehogegolven) = -inf;
end

q = belasting;
Z = sterkte - belasting;

```

## Stat\StatBerekenZGras

```

function [Z,q] = StatBerekenZGras (Profiel, ZSegment, WaarnStormIndex, ...
H_s, T_p, sl, theta_0)

%STATBEREKENZGRAS      In StatBerekenZGras wordt de Z-functie uitgerekend voor
%                      het faalmechanisme instabiliteit voor grasbekleding.
%
% AANROEP
% [Z,q] = BerekenZGras (Profiel, ZSegment, WaarnStormIndex, H_s, T_p, sl, theta_0)
% INVOER
% Profiel:      het dijksprofiel
% ZSegment:    karakteristiek van het dijksegment (ondergrens & bovengrens,
%              en bekledingsgegevens.
% WaarnStormIndex: de index van de storm behorend bij de waarneming
% H_s:         significante golfhoogte (nearshore)
% T_p:         de piekperiode (nearshore)
% sl:         waterstand (nearshore)
% theta_0:     golfrichting (nearshore) (t.o.v. het noorden, in radialen)
% UITVOER
% Z:          Z is gelijk aan de asfaltsterkte minus de belasting
% q:         de belasting q

% Auteur: Bert Jagers, WL | Delft Hydraulics

% 3-11-2003

%
% Bekledingsgegevens ...
%
grasmat      = ZSegment.bekledingsgegevens.kwaliteit; % 'goed','matig','slecht'
z_q         = StatBerekenOplooptniveau(Profiel, 0.1, H_s, T_p, sl, theta_0);

```



```

%
% Verticale locatie van het segment in het dijkprofiel ...
%
z_ondergrens = ZSegment.ondergrens;
z_bovengrens = ZSegment.bovengrens;
%
% Bepaal helling ...
%
helling = BepaalHelling(1.5,H_s,sl,Profiel);

sterkte = repmat(1,size(sl));
belasting = repmat(0,size(sl));

%
% Bereken ts/tsmax voor alle waarnemingen.
%
tijdstep = 1; % tijdstep van 1 uur

grasbovengolven = z_ondergrens > sl + z_q;
resterend = ~grasbovengolven;

ToetsGolfklappen = resterend & (z_ondergrens < sl);
ToetsStroming = resterend & ~ToetsGolfklappen;

%
% Toets op golfklappen ...
%
karakParam = 2*(H_s.^0.75).*(T_p.^0.5).*helling;
switch grasmatt
case 'goed'
    C1=1.0;
case 'matig'
    C1=0.7;
case 'slecht'
    C1=0.2;
end
belasting(ToetsGolfklappen & karakParam>=C1) = 2;
%belasting(ToetsGolfklappen & karakParam<0.2) = 0;
ToetsGolfklappen = ToetsGolfklappen & karakParam<C1 & karakParam>=0.2;

if ~strcmp(grasmatt,'slecht')
    switch grasmatt
    case 'goed'
        C2 = 2.3;
        C3 = -0.52;
    case 'matig'
        C2 = 1.6;
        C3 = -0.52;
    end
    maximale_belastingduur = (karakParam/C2).^(1/C3);
    belasting(ToetsGolfklappen) = tijdstep./maximale_belastingduur(ToetsGolfklappen);
end

%
% Toets op stroming ...
%
% STAP 1
%
z=z_ondergrens-sl;
%
g2pi=9.81/(2*pi);
rekensnelheid = 700*(H_s./T_p).*(0.085 - H_s./(g2pi*T_p.^2)).*sqrt(1-(z./z_q)).*helling;
%
%STAP 2
%
switch grasmatt
case 'goed'
    C1 = 1560;
    C2 = -4.8;
case 'matig'
    C1 = 276;
    C2 = -4.2;
case 'slecht'
    C1 = 50;
    C2 = -3.6;
end
maximale_belastingduur = C1*rekensnelheid.^C2;
%

```

```

%STAP 3
%
z=z_ondergrens-sl;
%
effectieve_tijdstap = (1 - z./z_q) * tijdstap;
%
%STAP 4
%
belasting(ToetsStroming) = ...
    effectieve_tijdstap(ToetsStroming) ./ maximale_belastingduur(ToetsStroming);

%
% Bepaal begin en eind van alle stormen in dataset en
% sommige belasting (=ts/tsmax) voor elke storm.
%
Start=cat(1,1,find(diff(WaarnStormIndex))+1,length(WaarnStormIndex)+1);
for storm = 1:length(Start)-1
    Storm=Start(storm):Start(storm+1)-1;
    belasting(Storm)=cumsum(belasting(Storm));
end

q = belasting;
Z = sterkte - belasting;

```

## Stat\StatBerekenZSteenzetting

```

function [Z,q] = StatBerekenZSteenzetting (Profiel, ZSegment, H_s, T_p, sl, theta_0)

%STATBEREKENZSTEENZETTING In StatBerekenZSteenzetting wordt de Z-functie
% uitgerekend voor het faalmechanisme instabiliteit voor steenzettingen.
% AANROEP
% [Z,q] = BerekenZSteenzetting (Profiel, Blok, H_s, T_p, sl, theta_0)
% INVOER
% Profiel: het dijkprofiel
% ZSegment: karakteristiek van het dijksegment (ondergrens & bovengrens,
% en bekledingsgegevens.
% H_s: significante golfhoogte (nearshore)
% T_p: de piekperiode (nearshore)
% sl: waterstand (nearshore)
% theta_0: golfrichting (nearshore) (t.o.v. het noorden, in radialen)
% UITVOER
% Z: Z is gelijk aan de sterkte van de steenzetting minus de
% belasting
% q: de belasting q

% Auteur: Bert Jagers, WL | Delft Hydraulics

% 4-11-2003

%
% Globale constanten ...
%
g2pi=9.81/(2*pi);
%
% Bekledingsgegevens ...
%
toplaagdikte = ZSegment.bekledingsgegevens.toplaagdikte;
filterkleidikte = ZSegment.bekledingsgegevens.filterkleidikte;
D15 = ZSegment.bekledingsgegevens.D15;
reldichtheid = ZSegment.bekledingsgegevens.relatievedichtheid;
bekledingstype = ZSegment.bekledingsgegevens.bekledingstype; %'normaal','ongunstig'
%
% Oploop niveau t.o.v. SWL ...
%
z_q = StatBerekenOplooptniveau(Profiel, 0.1, H_s, T_p, sl, theta_0);
%
% Verticale locatie van het segment in het dijkprofiel ...
%
z_ondergrens = ZSegment.ondergrens;
z_bovengrens = ZSegment.bovengrens;
%
% Bepaal helling ...
%
helling = BepaalHelling(1.5,H_s,sl,Profiel);
%

```

```

% Stap 0: Afschuiving
%
sterkte      = reldichtheid*toplaagdikte+filterkleidikte;
sterkte      = repmat(sterkte,size(sl));
%
% De volgende formule geeft de grens tussen GOED en TWIJFELACHTIG
%
belasting    = min(0.16*(H_s.^0.2).*(T_p.^1.6).*(helling.^0.8),1.5*H_s)-1334*D15*(1-
1.19*helling).*sqrt(T_p);
%
% Als sterkte > belasting, dan is de dijk GOED met betrekking tot
% afschuiving en moeten we dus kijken naar de top laag stabiliteit.
% Als sterkte < belasting, dan is de dijk TWIJFELACHTIG of ONVOLDOENDE
% en dus zullen we deze hier interpreteren als zijnde gefaald.
%
resterend    = sterkte > belasting;

%
% Stap 1: Toplaag stabiliteit
%
% Bepaal hoogte waar de golfklappen de dijkbekleding raken.
%
impacthoogte = sl - min(0.11*H_s.*(g2pi * (T_p.^2) .* helling ./ H_s).^0.8,1.5*H_s);
%
% Controleer of de locatie van golfklap of de effectieve golfoploop
% het dijksegment met de steenzetting omvat.
%
belastingbovensegment = impacthoogte > z_bovengrens;
belastingondersegment = sl + z_q/2 < z_ondergrens;
%
% Niet in bereik ---> gebruik de hierboven gedefinieerde sterkte en
% belasting betreffende het proces afschuiving.
%
resterend     = resterend & ~belastingbovensegment & ~belastingondersegment;
%
% Bij belasting rond de ondergrens van het dijksegment met steen-
% zetting vergroot de effectieve top laagdikte en ga door met de
% toets op top laagstabiliteit.
%
belastinggrondondergrens = sl < z_ondergrens & ~belastingondersegment;
toplaagdikte      = repmat(toplaagdikte,size(sl));
toplaagdikte(belastinggrondondergrens) = toplaagdikte(1)/0.8;

%
% Stap 2: Berekenen stabiliteitsgetal en golfbrekerparameter
%
stabiliteitsgetal = H_s./(reldichtheid*toplaagdikte);
golfbrekerparam   = helling./sqrt(H_s./(g2pi*T_p.^2));

%
% Stap 3: Bepaal de grenswaarde voor het stabiliteitsgetal
%
% De grenswaarde "n_s;onder" gebruikt hier is de grens tussen
% GOED en TWIJFELACHTIG. Deze grenswaarde wordt hier aangeduid
% als bovengrens_SG omdat het de bovengrens vormt voor de range
% van stabiliteitsgetallen waarin de top laag stabiel is.
%
% Er dient onderscheid gemaakt te worden tussen golfbreker-
% parameter (afgekort tot GBP) groter en kleiner dan 2.
%
GBPgroterdan2 = golfbrekerparam>2;
switch bekledingstype
case 'normaal'
    bovengrens_SG      = 4.08*golfbrekerparam.^(-1.014);
    bovengrens_SG(GBPgroterdan2) = 11*golfbrekerparam(GBPgroterdan2).^(-4) ...
    + 0.03*golfbrekerparam(GBPgroterdan2) ...
    + 1.25;
case 'ongunstig'
    bovengrens_SG      = 3.07*golfbrekerparam.^(-1.014);
    bovengrens_SG(GBPgroterdan2) = 6.5*golfbrekerparam(GBPgroterdan2).^(-4) ...
    + 0.02*golfbrekerparam(GBPgroterdan2) ...
    + 1.09;
end
sterkte(resterend) = bovengrens_SG(resterend) ./ max(stabiliteitsgetal(resterend),1e-20);
belasting(resterend) = 1;

%
% Uitvoer grootheden ...
    
```

```
%
q = belasting;
Z = sterkte - belasting;
```

## Stat\StatBerekenOplooptniveau

```
function z_q = StatBerekenOplooptniveau(Profiel, Qc, H_s, T_p, sl, theta_0)

%STATBEREKENOPLOOPTNIVEAU      In StatBerekenOplooptniveau wordt het oplooptniveau
%                               behorende bij het opgegeven debiet Qc uitgerekend gebruikmakend van de
%                               formules voor het faalmechanisme golfoverslag. Het eerste deel van de
%                               code komt overeen met de routine StatBerekenZGolfoverslag. Het tweede
%                               deel berekent vanuit het (overslag)debiet het bijbehorende oplooptniveau
%                               of de kritische kruinhoogte in plaats van uit de kruinhoogte het over-
%                               slaand debiet zoals StatBerekenZGolfoverslag dat berekent.
%
% AANROEP
%   z_q = StatBerekenOplooptniveau (Profiel, Qc, H_s, T_p, sl, theta_0)
% INVOER
%   Profiel:      het dijksprofiel
%   Qc:           het kritieke overslagdebiet (dat overslagdebiet dat nog
%               toelaatbaar is)
%   H_s:         significante golfhoogte (nearshore)
%   T_p:         de piekperiode of de spectrale periodemaat (nearshore)
%   sl:          waterstand (nearshore)
%   theta_0:     golfrichting (nearshore) (t.o.v. het noorden, in radialen)
% UITVOER
%   z_q:         het oplooptniveau (ook wel de kritieke overslag kruinhoogte h_k
%               bij het gespecificeerde kritieke overslagdebiet).
%
% EXTERNE FUNCTIES
%   StatBerekenZGolfoploop
%
% Auteur: Bert Jagers, WL | Delft Hydraulics
%
% Gebaseerd op StatBerekenZGolfoverslag (ref. versie Hydra-K 2.05)
% Copyright © 2000 Rijkswaterstaat/RIKZ All Rights Reserved.
%
% --- Vanaf hier tot aan soortgelijke eindmarker is de code identiek aan
% StatBerekenZGolfoverslag.
%
%
% Originele en nieuwe formulering hebben dezelfde functionele vorm,
% alleen de coëfficiënten verschillen enigszins. Vang ze daarom hier op.
%
if ~NieuweGolfFormules,
    Q_b_coeff = 0.06 ;
    R_b_coeff = 4.7 ;
else
    Q_b_coeff = 0.067 ;
    R_b_coeff = 4.3 ;
end

% Voor brekende golven geldt:
%  $q = Q_b * \gamma_b * \sqrt{g * H_s^3 * \tan(\alpha) / s_{op}}$ 
% waarbij:
% q = overslaand debiet [l/s/m]
% Q_b = dimensieloos overslagdebiet voor brekende golven
% gamma_b = reductiefactor voor invloed berm
% g = zwaartekrachtversnelling [m/s2]
% H_s = significante golfhoogte [m]
% alpha = helling van het talud [- is al een tangens]
% s_op = golfsteilheid

% Definitie golfsteilheid:
%  $s_{op} = 2 * \pi * H_s / (g * T_p^2)$ 
% waarbij:
% T_p = piekperiode [s]

% Definitie dimensieloos overslagdebiet bij brekende golven:
%  $Q_b = 0,06 * \exp(-4,7 * R_b)$ 
% waarbij:
% R_b = dimensieloze kruinhoogte bij brekende golven
```

```

% Definitie dimensieloze kruinhoogte bij brekende golven:
%  $R_b = h_k \cdot \sqrt{s_{op}} / (H_s \cdot \tan(\alpha) \cdot \gamma_b \cdot \gamma_f \cdot \dots$ 
%  $\gamma_{beta} \cdot \gamma_s \cdot \gamma_v)$ 
% waarbij:
%  $h_k$  = vrije kruinhoogte boven stilwaterlijn
%  $\gamma_f$  = reductiefactor voor invloed van ruwheid
%  $\gamma_{beta}$  = reductiefactor voor invloed van hoek van golfaanval
%  $\gamma_s$  = reductiefactor voor invloed van hoek van golfaanval
%  $\gamma_v$  = reductiefactor voor invloed van verticale wand

% Voor niet-brekende golven geldt:
%  $q = Q_n \cdot \sqrt{g \cdot H_s^3}$ 
% waarbij  $Q_n$  = maximum dimensieloos overslagdebiet bij niet-brekende golven

% Definitie dimensieloos overslagdebiet bij niet-brekende golven:
%  $Q_n = 0,2 \cdot \exp(-2,3 \cdot R_n)$ 
% waarbij:
%  $R_n$  = dimensieloze kruinhoogte bij niet-brekende golven

% Definitie dimensieloze kruinhoogte bij niet-brekende golven:
%  $R_n = h_k / (H_s \cdot \gamma_f \cdot \gamma_{beta} \cdot \gamma_s)$ 

hoogte_kruin = Profiel.hoogte_kruin;
theta_dijk = Profiel.theta_dijk * pi / 180;

g = 9.81;

h_k = hoogte_kruin - sl;
verschilhoek = abs(normaliseerHoek2(theta_0 - theta_dijk)) * 180/pi;
% normaliseerHoek0 normaliseert hoeken, zodat ze liggen tussen -pi en pi

[alfa, gamma_b, gamma_f, hoogte_2pr, s_op, zeta, z_2pr] ...
= StatBerekenZGolfoploop (Profiel, H_s, T_p, sl, theta_0);

% alfa = berekend door Golfoploop
% gamma_b = berekend door Golfoploop
% gamma_f = berekend door Golfoploop

max_gamma_beta = 0.736;
min_gamma_s = 0;
max_gamma_s = 1;
gamma_beta = max ( max_gamma_beta, 1 - verschilhoek .* 0.0033);
gamma_s = max ( min_gamma_s, min (max_gamma_s, 1 - (verschilhoek - 80) ./ 30));

% Dus
% gamma_beta = 0.736 als verschilhoek < 80
% gamma_beta = 1 - verschilhoek * 0,0022 als verschilhoek > 80

% Wat is de juiste waarde van gamma_v?
gamma_v = 1;

gamma_t = max(0.4, gamma_beta .* gamma_f .* gamma_b .* gamma_v);
gamma_totaal = gamma_t .* gamma_s;

mini = 1.e-10;
%
% --- Tot hier is de code identiek aan StatBerekenZGolfoverslag
%
%
% In plaats van het uitrekenen van het overslag debiet bij de
% kruinhoogte van het dijkprofiel reken terug vanuit het overslag
% debiet naar de minimale benodigde kruinhoogte.
%
%
% Bepaal de coëfficiënten voor de eerste relatie tussen z_q en q.
%
%  $\alpha_{b1} = 1000 \cdot \sqrt{g \cdot H_s^3} \cdot Q_{b\_coeff} \cdot zeta \cdot \gamma_b \cdot \sqrt{\alpha}$ ;
%  $\alpha_{b2} = -R_b\_coeff \cdot \max(\text{mini}, zeta \cdot H_s \cdot \max(\gamma_{totaal}, \text{mini}))$ ;
%
% Bepaal de coëfficiënten voor de tweede relatie tussen z_q en q.
%
%  $\alpha_{n1} = 0.2 \cdot 1000 \cdot \sqrt{g \cdot H_s^3}$ ;
%  $\alpha_{n2} = -2.3 \cdot \max(\text{mini}, H_s \cdot \gamma_s \cdot \gamma_f \cdot \gamma_{beta} \cdot \gamma_s)$ ;

```

```

%
% Bepaal voor beide vergelijkingen het niveau en neem het minimum.
%
z_qb = log(Qc./alfa_b1)./alfa_b2;
z_qn = log(Qc./alfa_n1)./alfa_n2;
z_q = min(z_qb,z_qn);

%
% Controle code: bereken overslaand debiet ...
%
%h_k = z_q;
%q_b = alfa_b1 .* exp(alfa_b2 .* h_k);
%q_n = alfa_n1 .* exp(alfa_n2 .* h_k);
%q = min(q_b,q_n);

```

## Stat\BerekenBekleding

```

function [paramWaarde,IndexFalen] = ...
    StatBerekenBekleding (Regio, Locatie, correctieRVB, WaarnOverschrFactor, Waarneming,
    WaarnStormSectorIndex, ...
    WaarnStormIndex, Stat, Profiel, HerhalingsTijd , Kappa , SelectieWaarnemingen, StroomCor,
    faalMechID, dijkopbouw)

%STATBEREKENBEKLEDING
% In StatBerekenBekleding wordt een eigenschap van de bekleding
% bepaald gegeven de herhalingstijd van falen en het dijkprofiel.
% Deze functie is conceptueel gelijk aan StatBerekenKruinhoogte.
%
%
% Voor continue parameters worden eerst een ondergrens en een
% bovengrens voor de te variëren parameter bepaald. Daarna wordt de
% benodigde waarde van de parameter bepaald met een minimalisatie-
% routine.
%
% Voor discrete parameters worden alle waarden doorlopen beginnend
% in het minimum.
%
% Bij elke waarde komt een aantal waarnemingen in het faalgebied.
% Het stopcriterium is als het aantal waarnemingen in het faal-
% gebied gelijk is aan kappa. (Kappa is eenduidig bepaald door de
% herhalingstijd.)
%
% AANROEP
% [paramWaarde,IndexFalen] = ...
%     StatBerekenBekleding (Regio, Locatie, correctieRVB, WaarnOverschrFactor, ...
%     Waarneming, WaarnStormSectorIndex, WaarnStormIndex, Stat, Profiel, ...
%     HerhalingsTijd , Kappa , SelectieWaarnemingen)
%
% INVOER
% Regio: string met naam van de regio
% Locatie: de x,y coördinaten van de locatie
% correctieRVB: de correctie op de waterstand voor verschillen met het
Randvoorwaardenboek
% WaarnOverschrFactor: formule: -log(waarnemingenoverschrijdingsaantallen/kappa)
% (waterstand, windsnelheid, piekperiode, sign. golfhoogte)
% Waarneming: de golfrichting en de windrichting van de waarnemingen
% WaarnStormSectorIndex: de sectorindex van de storm (windrichting van de storm)
% WaarnStormIndex: de index van de storm behorend bij de waarneming
% Freqbound: bovengrens voor de frequentie
% Stat: de Weibull parameters voor de waterstand, windsnelheid,
% piekperiode en sign. golfhoogte
% Profiel: het dijkprofiel
% Herhalingstijd: de ingevoerde herhalingstijd
% Kappa: het aantal verschoven waarnemingen in het faalgebied
% SelectieWaarnemingen: criterium voor de selectie van de waarnemingen: zie stap 2
% Methode de Haan op basis van de hoogste waterstand of op basis
% van de hoogste belasting
%
% UITVOER
% paramWaarde: de benodigde waarde van de parameter bij de herhalingstijd;
% de parameter is afhankelijk van de bekleding
% IndexFalen: een vector met de indices van de waarnemingen die in het
% faalgebied liggen
%
% EXTERNE FUNCTIES
% StatWeibullInv
% Offshore2Nearshore
% StatBerekenZuitNs
% SelecteerMaxZ
% StatKruinhBlokMinimizeFunc

```

```
% Copyright © 2000 Rijkswaterstaat/RIKZ All Rights Reserved.

% Auteur: Bert Jagers, WL | Delft Hydraulics

IndexFalen= [];

% bepalen offshorewaarden (waterstand met wind- en golfrichting)
offshoreWaarden= [StatWeibullInv(WaarnOverschrFactor+log(HerhalingsTijd),...
    WaarnStormSectorIndex,Stat,Regio) Waarneming];

% transformatie van offshore naar nearshore
nearshoreWaarden = Offshore2Nearshore ( Regio,Locatie.x,Locatie.y, correctieRVB,
offshoreWaarden,StroomCor,faalMechID);

faalMechanisme.naam = 'algemene instabiliteit';
faalMechanisme.dijkopbouw = dijkopbouw;

%
% Onderstaande gaat alleen goed als er maar 1 bekledingssegment is ....
%
discreteparameter = 0;
switch dijkopbouw.bekleding
case 'asfalt'
    ondergrens = 0.01;
    bovengrens = 50;
    parameter = 'laagdikte';
case 'blokken'
    ondergrens = 0.01;
    bovengrens = 50;
    parameter = 'dichtheid_hoog';
    faalMechanisme.dijkopbouw.bekledingsgegevens.blokdikte_laag = 1;
    faalMechanisme.dijkopbouw.bekledingsgegevens.blokdikte_hoog = 1;
case 'gras'
    parameter = 'kwaliteit';
    discreteparameter = 1;
    parameterwaarden = {'slecht','matig','goed'};
case 'steenzetting'
    ondergrens = 0.01;
    bovengrens = 50;
    parameter = 'relatievedichtheid';
    faalMechanisme.dijkopbouw.bekledingsgegevens.toplaagdikte = 1;
end

if discreteparameter
    for i = 1:length(parameterwaarden)
        faalMechanisme.dijkopbouw.bekledingsgegevens = ...
            setfield(faalMechanisme.dijkopbouw.bekledingsgegevens,parameter,parameterwaarden{i});
        %
        % Hier worden altijd alle waarnemingen (dus hele stormen voor zover relevant)
        % doorgegeven.
        %
        z = StatBerekenZuitNs (faalMechanisme, Profiel,nearshoreWaarden,WaarnStormIndex);
        [aantalZkleinerNul, Jm, zOorsprIndex] = ...
            SelecteerMaxZ ( z, Kappa, WaarnStormIndex, SelectieWaarnemingen);

        if aantalZkleinerNul <= Kappa,
            IndexFalen= zOorsprIndex(1:aantalZkleinerNul);
            paramWaarde = parameterwaarden{i};
            return
        end
    end
    %
    % zelf bovengrens voldoet niet...
    %
    IndexFalen = [];
    paramWaarde = Inf;
else % continue parameter
    faalMechanisme.ontwerpparameter = parameter;

    % kijk eerst of de ondergrens wel voldoende is
    faalMechanisme.dijkopbouw.bekledingsgegevens = ...
        setfield(faalMechanisme.dijkopbouw.bekledingsgegevens,parameter,ondergrens);
    if strcmp(parameter,'dichtheid_hoog')
        faalMechanisme.dijkopbouw.bekledingsgegevens.dichtheid_laag = ondergrens;
    end
end
```

```

z = StatBerekenZuitNs (faalMechanisme, Profiel,nearshoreWaarden,WaarnStormIndex);
[aantalZkleinerNul, Jm] = ...
    SelecteerMaxZ ( z, Kappa, WaarnStormIndex, SelectieWaarnemingen);

if aantalZkleinerNul < Kappa,
    % IndexFalen= zOorsprIndex(1:aantalZkleinerNul);
    IndexFalen= [];
    paramWaarde = -Inf;

    return
end

% kijk eerst of de bovengrens wel voldoende is
faalMechanisme.dijkopbouw.bekledingsgegevens = ...
    setfield(faalMechanisme.dijkopbouw.bekledingsgegevens,parameter,bovengrens);
if strcmp(parameter,'dichtheid_hoog')
    faalMechanisme.dijkopbouw.bekledingsgegevens.dichtheid_laag = bovengrens;
end

z = StatBerekenZuitNs (faalMechanisme, Profiel,nearshoreWaarden,WaarnStormIndex);
[aantalZkleinerNul, Jm] = ...
    SelecteerMaxZ ( z, Kappa, WaarnStormIndex, SelectieWaarnemingen);

if aantalZkleinerNul > Kappa,
    % IndexFalen= zOorsprIndex(1:aantalZkleinerNul);
    IndexFalen= [];
    paramWaarde = Inf;

    return
end

% de oplossing ligt ergens tussen ondergrens en bovengrens
% Nu de bovengrens zodanig maken dat aantalZkleinerNul < Kappa
%
% stapgrootte tussen minimum en maximum zodanig dat deze voor blokken
% ongeveer op 4 uitkomt net zoals in StatBerekenBloksterkte
%
stap = (bovengrens-ondergrens)/12.5;

while 1

    faalMechanisme.dijkopbouw.bekledingsgegevens = ...
        setfield(faalMechanisme.dijkopbouw.bekledingsgegevens,parameter,ondergrens+stap);
    if strcmp(parameter,'dichtheid_hoog')
        faalMechanisme.dijkopbouw.bekledingsgegevens.dichtheid_laag = ondergrens+stap;
    end

    z = StatBerekenZuitNs (faalMechanisme, Profiel,nearshoreWaarden,WaarnStormIndex);
    [aantalZkleinerNul, Jm] = ...
        SelecteerMaxZ ( z, Kappa, WaarnStormIndex, SelectieWaarnemingen);
    if aantalZkleinerNul > Kappa,
        ondergrens = ondergrens+stap;
    elseif aantalZkleinerNul == 0,
        stap = stap/4;
    else,
        break;
    end

end
bovengrens = ondergrens+stap;

count = 0;
stormduurafh = StormDuurAfhankelijk(faalMechanisme);
while 1

    if stormduurafh
        %
        % In het geval van stormduurafhankelijke bekleding moet de hele storm worden
        % doorgegeven ...
        %
        heleStorm=WaarnStormIndex==WaarnStormIndex(Jm);
        paramWaarde = fminbnd('StatKruinhBlokMinimizeFunc', ondergrens, bovengrens, ...
            optimset('TolX',1e-4,'Display','off'),...
            faalMechanisme, Profiel, nearshoreWaarden(heleStorm,:),WaarnStormIndex);
    else
        paramWaarde = fminbnd('StatKruinhBlokMinimizeFunc', ondergrens, bovengrens, ...
            optimset('TolX',1e-4,'Display','off'),...

```



```

        faalMechanisme, Profiel, nearshoreWaarden(Jm,:), WaarnStormIndex);
end

faalMechanisme.dijkopbouw.bekledingsgegevens = ...
    setfield(faalMechanisme.dijkopbouw.bekledingsgegevens, parameter, paramWaarde);
if strcmp(parameter, 'dichtheid_hoog')
    faalMechanisme.dijkopbouw.bekledingsgegevens.dichtheid_laag = paramWaarde;
end

count = count + 1;
if count > 100,
    Uitvoer('p', 'lus afgebroken');
    break;
end

z = StatBerekenZuitNs (faalMechanisme, Profiel, nearshoreWaarden, WaarnStormIndex);
[aantalZkleinerNul, Jm, zOorsprIndex] = ...
    SelecteerMaxZ ( z, Kappa, WaarnStormIndex, SelectieWaarnemingen);

if aantalZkleinerNul > Kappa,
    if abs(bovengrens-ondergrens) < 0.0001,
        % staat een beetje om Kappa heen te springen
        break;
    elseif abs(ondergrens-paramWaarde) < 0.0001,
        ondergrens = ondergrens+min(0.01*abs(ondergrens-bovengrens), 0.0005);
    else
        ondergrens= paramWaarde ;
    end

elseif aantalZkleinerNul < Kappa,
    if abs(bovengrens-paramWaarde) < 0.0001,
        bovengrens = bovengrens-min(0.01*abs(ondergrens-bovengrens), 0.0005);
    else
        bovengrens= paramWaarde ;
    end

else
    % hier is de oplossing gevonden
    break;
end

end

IndexFalen= zOorsprIndex(1:Kappa);
end

```

## Stat\private\BepaalHelling

```

function helling=BepaalHelling(a,H_s,sl,Profiel)
%BEPAAHELING Bepaal talud helling rekening houdend met berm.
%
% AANROEP
% helling = BepaalHelling(a,H_s,sl,Profiel)
% INVOER
% a:          vermenigvuldigingsfactor voor golfhoogte
% H_s:       significante golfhoogte (nearshore)
% sl:        waterstand (nearshore)
% Profiel:   het dijkprofiel
% UITVOER
% helling:   de representatieve helling
%
% In de buurt van de berm ... bepaal de horizontale afstand rekening
% houdend met de afstand tot de "berm". De berm zelf wordt niet mee-
% genomen in deze berekening: het is slechts een knikpunt.
%
% In de buurt van de berm is gedefinieerd als zijnde liggende tussen
% sl - a*H_s en sl + a*H_s.
%
helling_laag=Profiel.helling_laag;
helling_hoog=Profiel.helling_hoog;

verschil = sl - Profiel.hoogte_berm;
DX_laag = (a * H_s - verschil) ./ helling_laag;
DX_hoog = (a * H_s + verschil) ./ helling_hoog;
helling = 2 * a * H_s ./ (DX_laag + DX_hoog);

```

```
%
% Alleen op bovenste helling ... meer dan aH_s boven berm
%
helling (verschil > a * H_s) = helling_hoog;

%
% Alleen op onderste helling ... meer dan aH_s onder berm
%
helling (verschil > a * H_s) = helling_hoog;
```

## General\MeerdereBekledingen

```
function Ja = MeerdereBekledingen(UiWaarden)
%MEERDEREBEKLEDINGEN

Ja = 0;
if UiWaarden.faalmechanisme==4
    Ja = length(UiWaarden.dijkopbouw) > 1;
end
```

## General\printBekleding

```
function printBekleding (Waarden)

%PRINTPROFIEL      afdrukken van de bekleding
% AANROEP
%   printBekleding (Waarden)
% INVOER
%   Waarden:      structure-array met waarden van faalmechanisme,
%               profielen en bekleding

% EXTERNE FUNCTIES
%   Uitvoer

% Copyright © 2000 Rijkswaterstaat/RIKZ All Rights Reserved.

GlobalConstants

Bekleding = [];
if isfield(Waarden,'faalmechanisme') & Waarden.faalmechanisme == FMALGEMINSTAB
    Bekleding = Waarden.dijkopbouw;
elseif isfield(Waarden,'naam') & strcmp(Waarden.naam,'algemene instabiliteit')
    Bekleding = Waarden.dijkopbouw;
end
if ~isempty(Bekleding)
    Uitvoer('p','\n%18s\n','BEKLEDING');
    Bekleding = Waarden.dijkopbouw;
    Uitvoer('p','%25s%i\n','Aantal segmenten: ',length(Bekleding));
    for i=1:length(Bekleding)
        Uitvoer('p','%28s%i\n','SEGMENT: ',i);
        Uitvoer('p','%28s%s\n','Bekleding: ',upper(Bekleding(i).bekleding));
        Uitvoer('p','%28s%5.2f%s\n','Ondergrens: ',Bekleding(i).ondergrens,' m+NAP');
        Uitvoer('p','%28s%5.2f%s\n','Bovengrens: ',Bekleding(i).bovengrens,' m+NAP');
        gegevens = Bekleding(i).bekledingsgegevens;
        switch Bekleding(i).bekleding
            case 'asfalt'
                Uitvoer('p','%28s%5.2f%s\n','Laagdikte: ',gegevens.laagdikte,' m');
                Uitvoer('p','%28s%s\n','Ondergrondtype: ',gegevens.ondergrondtype);
                Uitvoer('p','%28s%s\n','Asfalttype: ',gegevens.asfalttype);
                Uitvoer('p','%28s%s\n','Filtertype: ',gegevens.filtertype);
            case 'blokken'
                Uitvoer('p','%28s%5.2f\n','Relatieve dichtheid: ',gegevens.dichtheid_hoog);
                Uitvoer('p','%28s%5.2f%s\n','Blokdikte: ',gegevens.blokdikte_hoog,' m');
            case 'gras'
                Uitvoer('p','%28s%s\n','Grasmat kwaliteit: ',gegevens.kwaliteit);
            case 'steenzetting'
                Uitvoer('p','%28s%5.2f%s\n','Toplaagdikte: ',gegevens.toplaagdikte,' m');
                Uitvoer('p','%28s%5.2f%s\n','Dikte filter- en kleilaag: ',gegevens.filterkleidikte,' m');

                Uitvoer('p','%28s%7.4f%s\n','D15 van zand: ',gegevens.D15,' m');
                Uitvoer('p','%28s%5.2f\n','Relatieve dichtheid: ',gegevens.relatievedichtheid);
                Uitvoer('p','%28s%s\n','Type constructie/bekleding: ',gegevens.bekledingstype);
```

```

end
end
end

```

## General\StormDuurAfhankelijk

```

function Ja = StormDuurAfhankelijk(UiWaarden)
%STORMDUURAFHANKELIJK

Ja = 0;
if isfield(UiWaarden,'faalm') & UiWaarden.faalm == 4
%
% Batch.invoer
%
Ja = 0.5;
elseif isfield(UiWaarden,'faalmechanisme') & UiWaarden.faalmechanisme == 4
%
% UiWaarden
%
Ja = 0.5;
elseif isfield(UiWaarden,'naam') & strcmp(UiWaarden.naam,'algemene instabiliteit')
%
% FaalMechanisme
%
Ja = 0.5;
end
if Ja == 0.5
Ja = 0;
AantalLagen = length(UiWaarden.dijkopbouw);
for i = 1:AantalLagen
if strcmp(UiWaarden.dijkopbouw(i).bekleding,'gras')
Ja = 1;
end
end
end
end

```

## Ui\CBbekleding

```

function Uitvoer = CBbekleding (Commando,Bestandsnaam)

%CBBEKLEDING          afhandelen callback van de pushbutton voor het
%                    wijzigen van de dijkbekleding voor de algemene
%                    (bekledings)instabiliteit.
%
% AANROEP EXTERN
%   CBbekleding ('SHOW')
%   Datastructuur = CBbekleding ('LEES',Bestandsnaam)
% AANROEP INTERN
%   CBbekleding (Commando)
% INVOER
%   Commando: Voor extern gebruik 'SHOW' en 'LEES', de overige commando's
%             zijn voor intern gebruik.
%   Bestandsnaam: In het geval van 'LEES' dient de aanroepende functie de
%                 naam van een dijkopbouwbestand mee te geven.
%   Datastructuur: In het geval van 'LEES' wordt een dijkopbouw structure
%                 teruggegeven.

% Auteur: Bert Jagers, WL | Delft Hydraulics

switch upper(Commando)
case 'SHOW'
ToonVenster(gcbo)
case 'CLOSE'
set(gcf,'closerequestfcn','closereq')
case 'SEGMENT'
SelecteerAnderSegment(gcbo)
case 'TOEVOEGEN'
ToevoegenSegment
case 'VERWIJDEREN'
VerwijderSegment
case 'BEKLEDING'
SelecteerAndereBekleding
case 'EDIT'
HaalData

```

```

case 'LEES'
    if isempty(gcbf)
        % Vanuit batch mode
        Uitvoer = LeesBekleding(Bestandsnaam);
    else
        % Vanuit user interface
        UiData = get(gcbf,'userdata');
        Dijkopbouw = LeesBekleding;
        if ~isempty(Dijkopbouw)
            UiData.dijkopbouw = Dijkopbouw;
        end
        set(gcbf,'userdata',UiData)
        VernieuwVenster(gcbf)
    end
case 'SCHRIJF'
    UiData = get(gcbf,'userdata');
    SchrijfBekleding(UiData.dijkopbouw);
otherwise
    fprintf('Onbekend commando: %s\n',Commando)
end

function SelecteerAndereBekleding
%
% Gebruiker heeft een ander type bekleding geselecteerd. Verander de
% eigenschappen van dit bekledingssegment overeenkomstig het nieuwe
% bekledingstype en update de user interface.
%
UiData = get(gcbf,'userdata');
i = get(UiData.handles.segmentscroller,'value');
bekledingStrings = get(UiData.handles.type,'string');
bekledingNr = get(UiData.handles.type,'value');
bekleding = bekledingStrings{bekledingNr};
%
% Als hetzelfde bekledingstype ook al elders op de dijk voorkomt, neem dan
% initieel dezelfde instellingen voor de bekledingsgegevens. Waarschijnlijk
% komen die instellingen beter overeen dan de default waarden die hier hard
% in de code staan.
%
gegevens = [];
for ii = 1:length(UiData.dijkopbouw)
    if strcmp(UiData.dijkopbouw(ii).bekleding,bekleding)
        gegevens = UiData.dijkopbouw(ii).bekledingsgegevens;
        break
    end
end
%
% Als hetzelfde bekledingstype nog niet elders op de dijk voorkomt, neem
% dan de onderstaande standaard waarden.
%
if isempty(gegevens)
    switch bekleding
    case 'asfalt'
        gegevens.laagdikte = 0.15;
        gegevens.ondergrondtype = 'klei';
        gegevens.asfalttype = 'waterbouwasfaltbeton';
        gegevens.filtertype = 'geotextiel';
    case 'blokken'
        gegevens.dichtheid_hoog = 1.6;
        gegevens.dichtheid_laag = 1.6;
        gegevens.blokdikte_hoog = 0.4;
        gegevens.blokdikte_laag = 0.4;
    case 'gras'
        gegevens.kwaliteit = 'goed';
    case 'steenzetting'
        gegevens.toplaagdikte = 0.4;
        gegevens.filterkleidikte = 3;
        gegevens.D15 = 0.001;
        gegevens.relatievedichtheid = 1.6;
        gegevens.bekledingstype = 'normaal';
    end
end
%
% Sla de nieuwe gegevens op in de datastructuur ...
%
UiData.dijkopbouw(i).bekleding = bekleding;
UiData.dijkopbouw(i).bekledingsgegevens = gegevens;
set(gcbf,'userdata',UiData);

```

```

%
% ... en update de user interface.
%
VernieuwVenster(gcbf)

function HaalData
%
% Lees data uit user interface objecten (met uitzondering van het nummer
% van het bekledingssegment en het bekledingstype).
%
UiData = get(gcbf,'userdata');
i = get(UiData.handles.segmentscroller,'value');
%
% Bepaal de locatie van het bekledingssegment op de dijk.
%
UiData.dijkopbouw(i).ondergrens =
HaalWaarde(UiData.handles.grenzen(2),UiData.dijkopbouw(i).ondergrens);
UiData.dijkopbouw(i).bovengrens =
HaalWaarde(UiData.handles.grenzen(5),UiData.dijkopbouw(i).bovengrens);
%
% Bereid het lezen van de bekledingsgegevens voor door indices in de
% UiData.handles array ...
%
switch UiData.dijkopbouw(i).bekleding
case 'asfalt'
    controls={'laagdikte',2
              'ondergrondtype',5
              'asfalttype',7
              'filtertype',9};
case 'blokken'
    controls={'dichtheid_hoog',2
              'blokdikte_hoog',5};
case 'gras'
    controls={'kwaliteit',2};
case 'steenetting'
    controls={'toplaagdikte',2
              'filterkleidikte',5
              'D15',8
              'relatieve dichtheid',11
              'bekledingstype',14};
end
%
% Generieke code om alle edit velden van het geselecteerde bekledingstype
% uit te lezen.
%
H = getfield(UiData.handles,UiData.dijkopbouw(i).bekleding);
gegevens = UiData.dijkopbouw(i).bekledingsgegevens;
for j = 1:size(controls)
    HuidigeWaarde = getfield(gegevens,controls{j,1});
    NieuweWaarde = HaalWaarde(H(controls{j,2}),HuidigeWaarde);
    gegevens = setfield(gegevens,controls{j,1},NieuweWaarde);
end
%
% Update data die afgeleid kan worden uit de gelezen data.
%
switch UiData.dijkopbouw(i).bekleding
case 'blokken'
    gegevens.dichtheid_laag = gegevens.dichtheid_hoog;
    gegevens.blokdikte_laag = gegevens.blokdikte_hoog;
end
UiData.dijkopbouw(i).bekledingsgegevens = gegevens;
set(gcbf,'userdata',UiData)
%
% Update de user interface (in het bijzonder de plot van het dijkprofiel).
%
VernieuwVenster(gcbf)

function Waarde = HaalWaarde(control,standaardWaarde)
%
% Hulpfunctie om de waarde van een enkele edit control in de user interface
% uit te lezen.
%
switch get(control,'style')
case 'edit'
    Waarde = sscanf(get(control,'string'),'%f',1);
    Grenzen = get(control,'userdata');

```

```

if ~isempty(Grenzen)
    if Waarde<Grenzen(1) | Waarde>Grenzen(2)
        Waarde = standaardWaarde;
    end
end
set(control, 'string', num2str(Waarde))
case 'popupmenu'
    Strings = get(control, 'string');
    i = get(control, 'value');
    Waarde = Strings{i};
end

function VerwijderSegment
%
% Verwijder het geselecteerde segment.
%
UiData = get(gcbf, 'userdata');
i = get(UiData.handles.segmentscroller, 'value');
UiData.dijkopbouw(i) = [];
set(gcbf, 'userdata', UiData);
%
% Kies een ander te editen segment. Pas de segmentscroller waarde aan; deze
% is maatgevend bij het updaten van de user interface.
%
AantalLagen = length(UiData.dijkopbouw);
if i>AantalLagen
    i = i-1;
end
set(UiData.handles.segmentscroller, 'max', AantalLagen, 'value', i)
%
% Update de user interface.
%
VernieuwVenster(gcbf)

function ToevoegenSegment
%
% Voeg een nieuw segment toe en laat dat een kopie zijn van het
% huidige segment.
%
UiData = get(gcbf, 'userdata');
i = get(UiData.handles.segmentscroller, 'value');
UiData.dijkopbouw(end+1) = UiData.dijkopbouw(i);
set(gcbf, 'userdata', UiData);
%
% Pas de range en de waarde van de segment scroller aan (deze is maatgevend
% bij het updaten van de user interface).
%
AantalLagen = length(UiData.dijkopbouw);
set(UiData.handles.segmentscroller, 'max', AantalLagen, 'value', AantalLagen)
%
% Update de user interface.
%
VernieuwVenster(gcbf)

function SelecteerAnderSegment(control)
%
% De gebruiker heeft een ander segment geselecteerd hetzij via het edit
% venster, hetzij via de segment scroller. Welke van de twee de gebruiker
% gebruikt heeft, wordt bepaald door het invoerargument "control".
%
UiData = get(gcbf, 'userdata');
if control==UiData.handles.segment
    Laag = sscanf(get(control, 'string'), '%i', 1);
else
    Laag = round(get(control, 'value'));
end
AantalLagen = length(UiData.dijkopbouw);
if Laag<1 | Laag>AantalLagen
    Laag = Data(1);
end
%
% Pas de segmentscroller aan (deze is maatgevend bij het updaten van de
% user interface).
%
set(UiData.handles.segmentscroller, 'max', AantalLagen, 'value', Laag)

```

```

%
% Update de user interface.
%
VernieuwVenster(gcbf)

function Dijkopbouw = LeesBekleding(Bestandsnaam)
%
% Lees bekleding van file. Bij een aanroep vanuit BatchRekenen wordt de
% bestandsnaam gespecificeerd. Vanuit de user interface vindt de aanroep
% plaats zonder specificatie van de bestandsnaam en wordt de gebruiker
% eerst om een bestandsnaam gevraagd.
%
if nargin < 1
    [bestand,pad] = uigetfile('*.bekleding');
    if ~ischar(bestand)
        Dijkopbouw = [];
        return
    end
    Bestandsnaam = fullfile(pad,bestand);
end
%
% Lees bestand ...
%
fid = fopen(Bestandsnaam,'r');
fgetl(fid); %HYDRA-K DIJKOPBOUWBESTAND
AantalLagen = sscanf(fgetl(fid),'%i',1); %i LAGEN
for i = 1:AantalLagen
    Dijkopbouw(i).bekleding = lower(strtok(fgetl(fid)));
    Dijkopbouw(i).ondergrens = sscanf(fgetl(fid),'%f',1); %f Ondergrens in m+NAP
    Dijkopbouw(i).bovangrens = sscanf(fgetl(fid),'%f',1); %f Bovengrens in m+NAP
    switch Dijkopbouw(i).bekleding
        case 'asfalt'
            gegevens.laagdikte = sscanf(fgetl(fid),'%f',1); %f Laagdikte in m
            gegevens.ondergrondtype = lower(strtok(fgetl(fid))); %s Ondergrondtype
            gegevens.asfalttype = lower(strtok(fgetl(fid))); %s Asfalttype
            gegevens.filtertype = lower(strtok(fgetl(fid))); %s Filtertype
        case 'blokken'
            gegevens.dichtheid_hoog = sscanf(fgetl(fid),'%f',1); %f Relatieve dichtheid
            gegevens.blokdikte_hoog = sscanf(fgetl(fid),'%f',1); %f Blokdikte in m
            gegevens.dichtheid_laag = gegevens.dichtheid_hoog;
            gegevens.blokdikte_laag = gegevens.blokdikte_hoog;
        case 'gras'
            gegevens.kwaliteit = lower(strtok(fgetl(fid))); %s Grasmat kwaliteit
        case 'steenzetting'
            gegevens.toplaagdikte = sscanf(fgetl(fid),'%f',1); %f Toplaagdikte in m
            gegevens.filterkleidikte = sscanf(fgetl(fid),'%f',1); %f Totale dikte van filter- en
            kleilaag in m
            gegevens.D15 = sscanf(fgetl(fid),'%f',1); %f D15 van het zand in m
            gegevens.relatievedichtheid = sscanf(fgetl(fid),'%f',1); %f Relatieve dichtheid
            gegevens.bekledingstype = lower(strtok(fgetl(fid))); %s Type constructie/bekleding
        otherwise
            fclose(fid);
            uiwait(msgbox(sprintf('Onbekend bekledingstype: '%s''.',Dijkopbouw(i).bekleding), ...
                'Foutmelding','modal'));
            Dijkopbouw = [];
            return;
    end
    Dijkopbouw(i).bekledingsgegevens = gegevens;
end
fclose(fid);

function SchrijfBekleding(Dijkopbouw)
%
% Schrijf bekleding naar file. Laat de gebruiker een bestandsnaam selecteren.
%
[bestand,pad] = uigetfile('*.bekleding');
if ~ischar(bestand)
    return
end
%
% Voeg de extensie .bekleding toe indien er geen extensie gespecificeerd
% is.
%
[p,b,e] = fileparts(bestand);
if isempty(e)
    bestand = [bestand '.bekleding'];
end

```

```

end
bestand = fullfile(pad,bestand);
%
% Schrijf het bestand ...
%
fid = fopen(bestand,'wt');
fprintf(fid,'HYDRA-K DIJKOPBOUWBESTAND\n');
AantalLagen = length(Dijkopbouw);
if AantalLagen == 1
    fprintf(fid,'1 LAAG\n');
else
    fprintf(fid,'%i LAGEN\n',AantalLagen);
end
for i = 1:AantalLagen
    fprintf(fid,'%s\n',Dijkopbouw(i).bekleding);
    fprintf(fid,'%f Ondergrens in m+NAP\n',Dijkopbouw(i).ondergrens);
    fprintf(fid,'%f Bovengrens in m+NAP\n',Dijkopbouw(i).bovangrens);
    gegevens = Dijkopbouw(i).bekledingsgegevens;
    switch Dijkopbouw(i).bekleding
        case 'asfalt'
            fprintf(fid,'%f Laagdikte in m\n',gegevens.laagdikte);
            fprintf(fid,'%s Ondergrondtype\n',gegevens.ondergrondtype);
            fprintf(fid,'%s Asfalttype\n',gegevens.asfalttype);
            fprintf(fid,'%s Filtertype\n',gegevens.filtertype);
        case 'blokken'
            fprintf(fid,'%f Relatieve dichtheid\n',gegevens.dichtheid_hoog);
            fprintf(fid,'%f Blokdikte in m\n',gegevens.blokdikte_hoog);
        case 'gras'
            fprintf(fid,'%s Grasmat kwaliteit\n',gegevens.kwaliteit);
        case 'steenzetting'
            fprintf(fid,'%f Toplaagdikte in m\n',gegevens.toplaagdikte);
            fprintf(fid,'%f Totale dikte van filter- en kleilaag in m\n',gegevens.filterkleidikte);
            fprintf(fid,'%f D15 van zand in m\n',gegevens.D15);
            fprintf(fid,'%f Relatieve dichtheid\n',gegevens.relatievedichtheid);
            fprintf(fid,'%s Type constructie/bekleding\n',gegevens.bekledingstype);
    end
end
fclose(fid);

function VernieuwGrafiek(UiFig)
%
% Vernieuw de plot van het dijkprofiel.
%
UiData = get(UiFig,'userdata');
segment = get(UiData.handles.segmentscroller,'value');
%
% Haal de constanten uit de structure.
%
breedte_berm = UiData.profiel(1).breedte_berm;

helling_laag = UiData.profiel(1).helling_laag;
helling_berm = UiData.profiel(1).helling_berm;
helling_hoog = UiData.profiel(1).helling_hoog;

hoogte_teen = UiData.profiel(1).hoogte_teen;
hoogte_berm = UiData.profiel(1).hoogte_berm;
hoogte_berm2 = hoogte_berm+max(breedte_berm*helling_berm,0.001);
hoogte_kruin = UiData.profiel(1).hoogte_kruin;

breedte_laag = (hoogte_berm-hoogte_teen)/helling_laag;
breedte_hoog = (hoogte_kruin-hoogte_berm2)/helling_hoog;
%
% Teken het dijkprofiel.
%
xx = cumsum([0 breedte_laag breedte_berm breedte_hoog]);
zz = [hoogte_teen hoogte_berm hoogte_berm2 hoogte_kruin];
area(xx,zz,hoogte_teen,'facecolor',[.8 .8 .8],'edgecolor','none','parent',UiData.assenstelsel);
%
% Voeg de teksten langs de assen toe.
%
set(UiData.assenstelsel,'ylim',[hoogte_teen hoogte_kruin])
set(get(UiData.assenstelsel,'xlabel'),'string','afstand (m) \rightarrow');
set(get(UiData.assenstelsel,'ylabel'),'string','hoogte (m+NAP) \rightarrow');
set(UiData.assenstelsel,'layer','top')

%
% Teken de bekledingssegmenten ... waarbij het actieve segment als een wat

```



```

% dikkere rode lijn wordt getekend.
%
AantalLagen = length(UiData.dijkopbouw);
Segment = UiData.dijkopbouw(segment);

for i = 1:AantalLagen
    z = [max(hoogte_teen,UiData.dijkopbouw(i).ondergrens) ...
        min(hoogte_kruin,UiData.dijkopbouw(i).bovengrens)];
    %
    % Houd rekening met de berm ...
    %
    if z(1) < hoogte_berm2 & hoogte_berm2 < z(2)
        z = [z(1) hoogte_berm2 z(2)];
    end
    if z(1) < hoogte_berm & hoogte_berm < z(2)
        z = [z(1) hoogte_berm z(2:end)];
    end
    %
    % Teken lijn ...
    %
    L = line(interp1(z,z,xx,z),z);
    if i == segment
        set(L,'color','r','linewidth',2)
    end
end

function VernieuwVenster(UiFig)
%
% Vernieuw het dialoogvenster.
%
UiData = get(UiFig,'userdata');
Actief = [1 1 1];
NietActief = get(UiFig,'color');
%
% Zet de selectie van het actieve bekledingssegment goed.
%
AantalLagen = length(UiData.dijkopbouw);
if AantalLagen == 1
    set(UiData.handles.segment,'string','1','enable','inactive', ...
        'backgroundcolor',NietActief,'tooltip','')
    set(UiData.handles.segmentscroller,'value',1,'max',2,'enable','off')
    set(UiData.handles.segmentverwijder,'enable','off');
    segment = 1;
else
    segment = get(UiData.handles.segmentscroller,'value');
    if segment > AantalLagen
        segment = 1;
    end

set(UiData.handles.segment,'string',num2str(segment),'enable','on','backgroundcolor',Actief,'tool
tip',sprintf('1 tot %i',AantalLagen))

set(UiData.handles.segmentscroller,'value',segment,'max',AantalLagen,'enable','on','sliderstep',[
min(1/(AantalLagen-1),0.9) 1])
set(UiData.handles.segmentverwijder,'enable','on');
end
%
% Selecteer het betreffende bekledingssegment.
%
Segment = UiData.dijkopbouw(segment);
%
% Toon de locatie en het bekledingstype
%
set(UiData.handles.grenzen(2),'string',num2str(Segment.ondergrens))
set(UiData.handles.grenzen(5),'string',num2str(Segment.bovengrens))
bekledingstypen = get(UiData.handles.type,'string');
set(UiData.handles.type,'value',strmatch(Segment.bekleding,bekledingstypen))
%
% Verberg de edit objecten voor de niet geselecteerde bekledingstypes.
%
for nietgeselecteerdeBekleding = setdiff(bekledingstypen,Segment.bekleding)
    set(getfield(UiData.handles,nietgeselecteerdeBekleding{1}),'visible','off')
end
set(getfield(UiData.handles,Segment.bekleding),'visible','on')
%
% Toon voor het geselecteerde bekledingstype de edit objecten.
%

```

```

switch Segment.bekleding
case 'asfalt'
    controls={'laagdikte',2
            'ondergrondtype',5
            'asfalttype',7
            'filtertype',9};
case 'blokken'
    controls={'dichtheid_hoog',2
            'blokdikte_hoog',5};
case 'gras'
    controls={'kwaliteit',2};
case 'steenzetting'
    controls={'toplaagdikte',2
            'filterkleidikte',5
            'D15',8
            'relatievedichtheid',11
            'bekledingstype',14};
end
H = getfield(UiData.handles,Segment.bekleding);
for j = 1:size(controls)
    HuidigeWaarde = getfield(Segment.bekledingsgegevens,controls{j,1});
    Hcontrol = H(controls{j,2});
    switch get(Hcontrol,'style')
    case 'edit'
        set(Hcontrol,'string',num2str(HuidigeWaarde));
    case 'popupmenu'
        Strings = get(Hcontrol,'string');
        Waarde = strmatch(HuidigeWaarde,Strings);
        set(Hcontrol,'value',Waarde);
    end
end
%
% Vernieuw de plot van het dijkprofiel.
%
VernieuwGrafiek(UiFig)

function ToonVenster(PushButtonHandle)
%TOONVENSTER Toon het edit venster voor dijkbekleding.
%
% Initialisatie van het venster en vraag het profiel en de bekleding op.
%
parent = get(PushButtonHandle,'Parent');
uiWaarden = FCHydra_k('WAARDEN',parent);
UiFig = figure('numbertitle','off','integerhandle','off','name','Hydra-K
bekleding','menubar','none','visible','off','resize','off');
set(UiFig,'position',get(UiFig,'position')+[0 0 200 0]);
UiData.assenstelsel = axes('units','normalized','position',[.48 .1 .50 .86]);
UiData.profiel = uiWaarden.profiel;
UiData.dijkopbouw = uiWaarden.dijkopbouw;
%
% Initialiseer de weergave van de edit objecten.
%
set(UiFig,'units','points');
UiFigPositie = get(UiFig,'position');
voffset = UiFigPositie(4)-30;
%
% Lees en schrijf knoppen (het is niet mogelijk om in een "modal"
% dialoogvenster menu's weer te geven, vandaar dat we knoppen gebruiken).
%
set(maakTextControl(UiFig,'right',[10 voffset 75 15],''),'string','Bestand:')
ph=maakPushButtonControl(UiFig,[90 voffset],[],'CBbekleding LEES','Lees bekleding uit een
bestand');
set(ph,'string','Lees','position',get(ph,'position')-[0 0 4 0])
ph=maakPushButtonControl(UiFig,[140 voffset],[],'CBbekleding SCHRIJF','Schrijf bekleding naar een
bestand');
set(ph,'string','Schrijf','position',get(ph,'position')-[0 0 4 0])
%
% Objecten voor de selectie van het bekledingssegment (edit venster en
% scroller).
%
voffset = voffset-30;
set(maakTextControl(UiFig,'right',[10 voffset 75 15],''),'string','Bekledingssegment:')
UiData.handles.segment = maakEditControl(UiFig,'right',[90 voffset 45 15],'CBbekleding
SEGMENT',1,'Selecteer het bekledingssegment');
UiData.handles.segmentscroller =
uicontrol('style','slider','parent',UiFig,'units','points','position',[135 voffset 10
15],'min',1,'max',2,'value',1,'enable','off','callback','CBbekleding SEGMENT');

```

```

%
% Toevoeg en verwijder knoppen voor bekledingssegmenten.
%
voffset = voffset-18;
ph=maakPushbuttonControl(UiFig,[90 voffset],[],'CBbekleding TOEVOEGEN','Voeg een segment toe');
set(ph,'string','Toevoegen','position',get(ph,'position')-[0 0 4 0])
ph=maakPushbuttonControl(UiFig,[140 voffset],[],'CBbekleding VERWIJDEREN','Verwijder dit
segment');
set(ph,'string','Verwijderen','position',get(ph,'position')-[0 0 4 0])
UiData.handles.segmentverwijder=ph;
%
% Positie van de bekleding op het profiel.
%
voffset = voffset-30;
set(maakTextControl(UiFig,'left',[10 voffset 120 13],'A'),'string','Positie van bekleding op
dijk:')
velden={'Ondergrens:','Ondergrens in m+NAP',[],'m+NAP'
'Bovengrens:','Bovengrens in m+NAP',[],'m+NAP'};
[UiData.handles.grenzen,voffset] = maakEditVelden (UiFig,voffset,velden,'on');
%
% Bekledingstype selectie lijst.
%
voffset = voffset-30;
set(maakTextControl(UiFig,'right',[10 voffset 75 13],''),'string','Bekledingstype:')
UiData.handles.type =
uiControl('style','popupmenu','parent',UiFig,'units','points','position',[90 voffset 100
15],'string',{'asfalt','blokken','gras','steen-zetting'],'backgroundcolor','w','callback','CBbekte
ding BEKLEDING');
voffsetref = voffset;
%
% Bekledingstype specifieke gegevens.
%
%----- ASFALT
voffset = voffsetref-15;
velden={'Laagdikte:','',[],'m'};
[H1,voffset] = maakEditVelden (UiFig,voffsetref-15,velden,'off');
velden={'Ondergrondtype:',{'zand','klei'}
'Asfalttype:',{'waterbouwasfaltbeton','opensteen-asfalt'],'asfalttype',
'Filtertype:',{'geotextiel','zand-asfalt'],'filtertype alleen relevant in het geval van
opensteen-asfalt'};
H2 = maakPopupMenuVelden (UiFig,voffset,velden,'off');
UiData.handles.asfalt = cat(2,H1,H2);
%
%----- BLOKKEN
voffset = voffsetref-15;
velden={'Relatieve dichtheid:','1 <= dichtheid <= 5',[1 5],'
'Blokdikte:','0 <= blokdikte <= 10',[0 10],'m'};
UiData.handles.blokken = maakEditVelden (UiFig,voffset,velden,'off');
%
%----- GRAS
voffset = voffsetref-15;
velden={'Kwaliteit:',{'goed','matig','slecht'],'kwaliteit van de grasmat'};
UiData.handles.gras = maakPopupMenuVelden (UiFig,voffset,velden,'off');
%
%----- STEENZETTING
voffset = voffsetref-15;
velden={'Toplaagdikte:','',[],'m'
'Dikte filter+kleilaag:','',[],'m'
'D15 van zand:','',[],'m'
'Relatieve dichtheid:','1 <= dichtheid <= 5',[1 5],''};
[H1,voffset] = maakEditVelden (UiFig,voffset,velden,'off');
velden={'Bekledingstype:',{'normaal','ongunstig'],'kwaliteit van de steenzetting'};
H2 = maakPopupMenuVelden (UiFig,voffset,velden,'off');
UiData.handles.steen-zetting = cat(2,H1,H2);
%-----
%
% Toon venster en wacht op venster afsluiting.
%
set(UiFig,'userdata',UiData,'windowstyle','modal','closerequestfcn','CBbekleding
CLOSE','visible','on');
VernieuwVenster(UiFig)
waitfor(UiFig,'closerequestfcn')
UiData = get(UiFig,'userdata');
delete(UiFig);
if ~isempty(UiData)
    set(PushButtonHandle,'userdata',UiData.dijkopbouw)
end
    
```

```

function [H,voffset] = maakEditVelden (UiFig,voffset,velden,visible)
% MAAKEDITVELDEN
%
% Hulpfunctie voor het tonen van edit controls.
%
b = 0;
for i = 1:size(velden,1)
    voffset = voffset-15;
    H(b+1) = maakTextControl(UiFig,'right',[10 voffset 75 13],'');
    set(H(b+1),'string',velden{i,1},'visible',visible)
    H(b+2) =      maakEditControl(UiFig,'right',[90      voffset      45      15],'CBbekleding
EDIT',velden{i,3},velden{i,2});
    set(H(b+2),'visible',visible)
    H(b+3) = maakTextControl(UiFig,'left',[140 voffset 55 13],'');
    set(H(b+3),'string',velden{i,4},'visible',visible)
    b=b+3;
end

```

```

function [H,voffset] = maakPopupmenuVelden (UiFig,voffset,velden,visible)
% MAAKLISTVELDEN
%
% Hulpfunctie voor het tonen van popupmenu controls.
%
b = 0;
for i = 1:size(velden,1)
    voffset = voffset-15;
    H(b+1) = maakTextControl(UiFig,'right',[10 voffset 75 13],'');
    set(H(b+1),'string',velden{i,1},'visible',visible)
    H(b+2) = uicontrol('parent',UiFig,'style','popupmenu', ...
        'horizontalalignment','left','string',velden{i,2}, ...
        'units','points','position',[90 voffset 100 15],'visible',visible, ...
        'backgroundcolor','w','tooltip',velden{i,3}, ...
        'callback','CBbekleding EDIT');
    b=b+2;
end

```



## **WL | Delft Hydraulics**

Rotterdamseweg 185  
postbus 177  
2600 MH Delft  
telefoon 015 285 85 85  
telefax 015 285 85 82  
e-mail [info@wldelft.nl](mailto:info@wldelft.nl)  
internet [www.wldelft.nl](http://www.wldelft.nl)

Rotterdamseweg 185  
p.o. box 177  
2600 MH Delft  
The Netherlands  
telephone +31 15 285 85 85  
telefax +31 15 285 85 82  
e-mail [info@wldelft.nl](mailto:info@wldelft.nl)  
internet [www.wldelft.nl](http://www.wldelft.nl)

