

Handreiking Hydra-Ring

Macrostablieiteit binnenwaarts



Handreiking Hydra-Ring

Macrostabiliiteit binnenwaarts

Auteur(s)

Karolina Wojciechowska

Robert Leander

Handreiking Hydra-Ring

Macrostablieit binnenwaarts




Opdrachtgever	Rijkswaterstaat Water, Verkeer en Leefomgeving
Contactpersoon	Robert Slomp
Referenties	-
Trefwoorden	Handreiking, Hydra-Ring, macrostablieit binnenwaarts, fragility curves, JSON, SQL

Documentgegevens

Versie	1.0
Datum	29-09-2021
Projectnummer	11206818-030
Document ID	11206818-030-GEO-0001
Pagina's	31
Classificatie	
Status	definitief

Auteur(s)

	Karolina Wojciechowska	
	Robert Leander	

Doc. Versie	Auteur	Controle	Akkoord	Publicatie
1.0	Karolina Wojciechowska 	David Rodriguez Aguilera 	Jan Aart van Twillert 	

Samenvatting

Hydra-Ring is de probabilistische rekenhart van Riskeer. Met Hydra-Ring worden de hydraulische belastingen en faalkansen voor enkele toetssporen afgeleid.

Recent is het faalmechanisme *Macrostabiliteit binnenwaarts* aan Hydra-Ring toegevoegd. In de gekozen aanpak worden eerst conditionele kansen op macrostabiliteit afgeleid met D-Stability voor een reeks van lokale waterstanden. De resultaten van D-Stability worden in de vorm van fragility curves aan Hydra-Ring doorgegeven, waarin de integratie met de belasting plaatsvindt. Hydra-Ring berekent de faalkans per jaar voor het faalmechanisme.

Deze functionaliteit van Hydra-Ring is nog niet in Riskeer beschikbaar. Dit document beschrijft het uitvoeren van probabilistische berekeningen met Hydra-Ring voor het faalmechanisme macrostabiliteit binnenwaarts (zonder Riskeer).

Het uitvoeren van de berekeningen vereist een aantal voorbereidingsstappen, waarvan de belangrijkste is het vertalen van de resultaten van D-Stability (fragility curves) naar invoer voor Hydra-Ring. Hiervoor is een speciale Python script ontwikkeld. Het werken van de script wordt ook in dit document behandeld.

Inhoud

	Samenvatting	4
1	Inleiding	7
1.1	Leeswijzer	7
1.2	Doelgroep	7
1.3	Appendix	7
2	Uitvoeren van berekeningen met Hydra-Ring	9
2.1	Berekeningen opstarten	10
2.1.1	INI-bestand	10
2.1.2	BAT-bestand	10
2.1.3	Starten van Hydra-Ring	11
3	Genereren van het SQL-bestand	12
3.1	Conversie JSON-bestanden naar SQL-regels	12
3.1.1	Eisen aan de D-Stability uitvoer	12
3.1.2	De conversie aanroepen vanaf de commando-regel	13
3.1.3	De conversie aanroepen vanuit de gebruikers grafische omgeving (GUI)	13
3.1.4	Uitvoer van het conversie script	15
3.2	Klaarzetten SQL-bestand	16
3.2.1	HydraulicModels	16
3.2.2	Sections	17
3.2.3	DesignTables	17
3.2.4	Numerics	17
3.2.5	VariableDatas	18
3.2.6	CalculationProfiles	18
3.2.7	SectionFaultTreeModels	19
3.2.8	SectionSubMechanismModels	19
3.2.9	ForelandModels	19
3.2.10	Forelands	19
3.2.11	Breakwaters	20
4	Inspectie van Hydra-Ring resultaten	21
5	Referenties	24
A	Voorbeeld van SQL-bestand	25
B	Opzoeken van LocationId	28
B.1	In Riskeer	28
B.2	Direct in database	28
C	Omrekenen betrouwbaarheidsindex ↔ faalkans	30
C.1	Van betrouwbaarheidsindex naar faalkans	30

1 Inleiding

Hydra-Ring is de probabilistische rekenhart van Riskeer. Met Hydra-Ring worden de hydraulische belastingen en faalkansen voor enkele toetssporen afgeleid.

Recent is het faalmechanisme *Macrostabiliiteit binnenwaarts* aan Hydra-Ring toegevoegd. In de gekozen aanpak worden eerst conditionele kansen op macrostabiliiteit afgeleid met D-Stability voor een reeks van lokale waterstanden (en een gegeven overslagdebiet). De resultaten van D-Stability worden in de vorm van fragility curves aan Hydra-Ring doorgegeven, waarin de integratie met de belasting plaatsvindt (Spee, 2020). Hydra-Ring berekent de faalkans per jaar voor het faalmechanisme.

De berekeningen met Hydra-Ring kunnen voor twee situaties worden uitgevoerd:

- **Eén fragility curve:** in de berekeningen wordt over de lokale waterstand geïntegreerd;
- **Meerdere fragility curves:** in deze situatie hoort elke fragility curve bij een bepaald overslagdebiet; Hydra-Ring integreert de informatie over de lokale waterstand en golfcondities (per windrichting). In dit geval is het dus belangrijk om het overslagdebiet per fragility curve te kunnen inschatten.

Deze functionaliteit van Hydra-Ring is nog niet in Riskeer beschikbaar. Dit document beschrijft het uitvoeren van probabilistische berekeningen met Hydra-Ring voor het faalmechanisme macrostabiliiteit binnenwaarts (zonder Riskeer).

De berekeningen hebben betrekking op één doorsnede/vak en één ondergrondscenario. Het uitvoeren van de berekeningen vereist een aantal voorbereidingsstappen, waarvan de belangrijkste is het vertalen van de resultaten van D-Stability (fragility curves) naar invoer voor Hydra-Ring. Hiervoor is een speciale Python script ontwikkeld. Het Python script vertaalt JSON-bestanden uit D-Stability naar een reeks van SQL-regels, die in een SQL-bestand opgenomen worden. Naast de informatie uit de fragility curves, bevat een volledig SQL-bestand ook rekeninstellingen, geometrie ten behoeve van overslagberekeningen en het verwijst naar de hydraulische locatie. Gegeven het SQL-bestand, het INI-bestand en de hydraulische database kunnen Hydra-Ring berekeningen vanaf de commando-regel gestart worden.

1.1 Leeswijzer

In Hoofdstuk 2 wordt het aanzetten van Hydra-Ring toegelicht. Hoofdstuk 3 presenteert het genereren van het SQL-bestand inclusief de conversie van JSON-bestanden met behulp van het Python script. Het inspecteren van de resultaten van Hydra-Ring wordt in Hoofdstuk 4 gepresenteerd. Bijlage A geeft een voorbeeld van een volledig SQL-bestand weer. In Bijlage B wordt het uitzoeken van de id van een hydraulische locatie toegelicht. Bijlage C legt uit het omrekenen van een betrouwbaarheidsindex naar de faalkans toe (en vice versa).

1.2 Doelgroep

De doelgroep van dit document zijn beoordelaars van dijken die bekend zijn met D-Stability en het idee van probabilistisch rekenen.

1.3 Appendix

Dit document gaat gepaard met de volgende bestanden:

- Map met de Python script "ConvertDStabilitytoSQL", met de volgende bestanden erin:

- dstabfragility.py
 - dstabfragility_cli.py
 - dstabfragility_gui.py
 - mkCD.py
- Templates voor:
 - SQL-bestand
 - INI-bestand
 - BAT-bestand

2 Uitvoeren van berekeningen met Hydra-Ring

Hydra-Ring gebruikt meerdere invoer bestanden, waaronder de belangrijkste zijn:

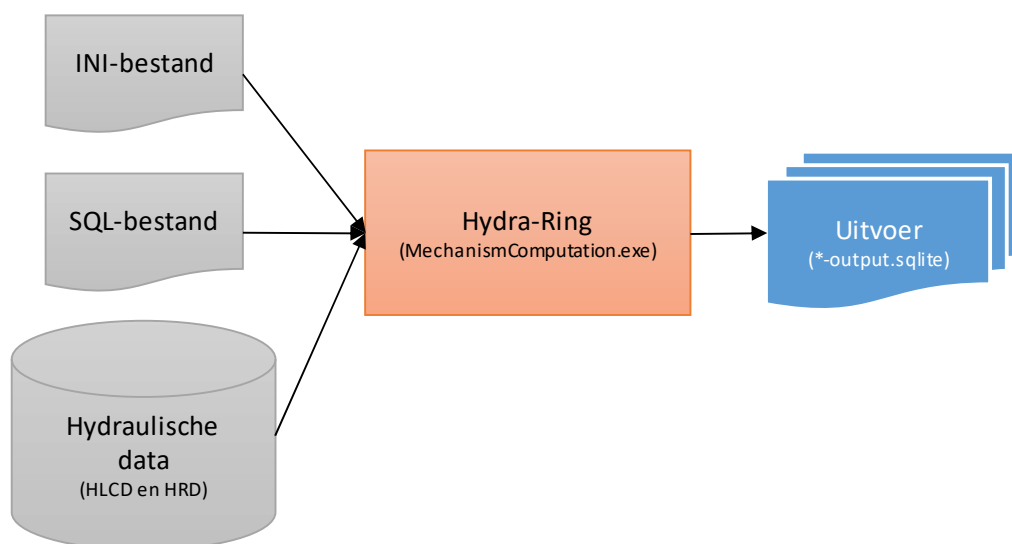
- SQL-bestand met o.a. hydraulische belastingenlocatie, rekeninstellingen, stochasten, dijkprofielen.
- Hydraulische data bestaande uit een HLCD bestand (met de statistiek) en een HRD bestand (met de lokale waterstanden en golfcondities).
- INI-bestand met o.a. de keuze van het faalmechanisme en paden naar andere benodigde bestanden. In het INI-bestand staat o.a. een verwijzing naar het SQL-bestand en de hydraulische data (hlcd.sqlite).

Hydra-Ring wordt via een executable (*MechanismComputation.exe*) aangeroepen. De executable kan vanaf de commando-regel gestart worden met het INI-bestand als de invoer.

De uitvoer van Hydra-Ring bestaat uit een reeks van bestanden:

- **-input.txt* met een overzicht van de gebruikte inputs.
- **-output.sqlite* met de tussen- en eindresultaten.
- *hydraring.log* met een overzicht van alle berichten (ook waarschuwingen) die tijdens het uitvoeren van berekeningen door Hydra-Ring zijn gegenereerd.
- *HydraRingLog.txt*, een tekstbestand met een samenvatting van de tussenresultaten (niet relevant hier).
- *combin.sqlite* met invoer t.b.v. het bepalen van de overstromingskans van het dijktraject (niet relevant hier).

In het onderstaande figuur wordt het uitvoeren van Hydra-Ring berekeningen schematisch weergegeven. In de paragraaf eronder wordt het aanzetten van Hydra-Ring berekeningen verder toegelicht. Het genereren van het SQL-bestand voor het faalmechanisme macrostabiliteit binnenwaarts wordt in Hoofdstuk 3 beschreven.



2.1 Berekeningen opstarten

De volgende bestanden zijn nodig om Hydra-Ring berekeningen op te starten voor één locatie (bijv. langs traject 48-1):

- SQL-bestand beschreven in Hoofdstuk 3 (met als naam bijv. *test.sql*).
- INI-bestand (met als naam bijv. *iniHydraRing.ini*).
- BAT-bestand (met als naam bijv. *runHydraRing.bat*).
- Map met Hydra-Ring bestanden (met als naam bijv. *HydraRing*), te verkrijgen via de Helpdesk water.
- Map met Hydraulische data (*WBI2017_Bovenrijn_48-1_v04*), te downloaden van de web-omgeving van Helpdesk water.

Hieronder volgt een beschrijving van de INI- en BAT-bestanden, en worden de stappen beschreven die nodig zijn om Hydra-Ring berekeningen op te starten.

2.1.1 INI-bestand

In het INI-bestand dienen de blauw gemarkeerde velden te worden aangepast. Het eerste gemarkeerde veld geeft de naam van het SQL-bestand weer. De tweede en derde velden geven de paden naar respectievelijk de map met Hydra-Ring en de map met Hydraulische data weer.

```
section = 1
mechanism = 102
alternative = 1
layer = 1
logfile = 1.log
outputverbosity = basic
outputtofile = file
projectdbfilename = test.sql
configdbfilename = C:\Voorbeeld\HydraRing\config.sqlite
hydraulicdbfilename = C:\Voorbeeld\WBI2017_Bovenrijn_48-1_v04\hlcd.sqlite
designpointOutput = sqlite
```

2.1.2 BAT-bestand

Het BAT-bestand wordt gebruikt om Hydra-Ring sommen op te starten. Daarnaast zorgt dit bestand dat de resultaten naar de map "Resultaten" verplaatst worden na het uitvoeren van de berekeningen. In het BAT-bestand dient eventueel het blauw gemarkeerde pad naar de Hydra-Ring map te worden aangepast. Ook de naam van het INI-bestand (hier *iniHydraRing.ini*) kan naar wens worden aangepast.

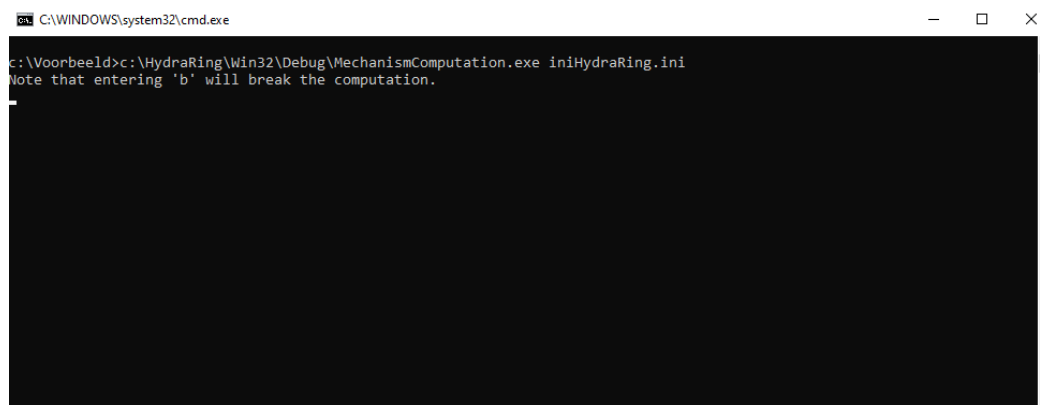
```
c:\Voorbeeld\HydraRing\MechanismComputation.exe iniHydraRing.ini
move *.sqlite Resultaten
move *.txt Resultaten
move *.log Resultaten
```

2.1.3 Starten van Hydra-Ring

Het aanzetten van een Hydra-Ring berekening wordt aan de hand van het onderstaande voorbeeld toegelicht. Alle uitgedrukte namen zijn voorbeelden.

Het proces bestaat uit de volgende stappen:

1. Maak een nieuwe map (*C:\Voorbeeld*) met de volgende bestanden/mappen erin:
 - o SQL-bestand: *test.sql*
 - o INI-bestand: *iniHydraRing.ini*
 - o BAT-bestand: *runHydraRing.bat*
 - o Map met de hydraulische data "*WBI2017_Bovenrijn_48-1_v04*" (met alle bestanden erin)
 - o Map met Hydra-Ring "*HydraRing*" (met alle bestanden erin)
 - o Map "*Resultaten*"
2. Zorg dat de INI- en BAT-bestanden de juiste informatie bevatten.
3. Voer de berekening uit door op het BAT-bestand dubbel te klikken.
4. Er verschijnt het volgende command prompt-scherm:



```
C:\WINDOWS\system32\cmd.exe
c:\Voorbeeld>c:\HydraRing\Win32\Debug\MechanismComputation.exe iniHydraRing.ini
Note that entering 'b' will break the computation.
```

5. Dit scherm wordt automatisch afgesloten na het uitvoeren van de berekening.
6. De resultaten van de berekening zijn in de map "Resultaten" te vinden.

Merk op dat er meerdere Hydra-Ring berekeningen achterelkaar aangezet kunnen worden, door het BAT-bestand uit te breiden (zie hieronder). De INI-bestanden bevatten in dat geval telkens nieuwe gegevens (en verwijzingen naar andere SQL-bestanden).

```
c:\Voorbeeld\HydraRing\MechanismComputation.exe iniHydraRing_1.ini
move *.sqlite Resultaten_1
move *.txt Resultaten_1
move *.log Resultaten_1

c:\Voorbeeld\HydraRing\MechanismComputation.exe iniHydraRing_2.ini
move *.sqlite Resultaten_2
move *.txt Resultaten_2
move *.log Resultaten_2
```

3 Genereren van het SQL-bestand

In dit hoofdstuk wordt het genereren van het SQL-bestand voor het faalmechanisme macrostabiliteit binnenwaarts toegelicht. Naast de informatie uit de fragility curves, bevat een volledig SQL-bestand ook rekeninstellingen, geometrie ten behoeve van overslagberekeningen en het verwijst naar de hydraulische locatie.

Het bestand bestaat uit een aantal tabellen (SQL-regels). Als het om het faalmechanisme macrostabiliteit binnenwaarts gaat hebben enkele tabellen betrekking op fragility curves. Deze net genoemde tabellen kunnen met behulp van een Python script automatisch gegenereerd worden, zoals beschreven in paragraaf 3.1. De overige tabellen zijn algemeen, zie paragraaf 3.2 voor verdere toelichting. Een voorbeeld van een volledig SQL-bestand staat in Bijlage A.

3.1 Conversie JSON-bestanden naar SQL-regels

D-Stability kan JSON-bestanden met fragility curves (en de bijbehorende informatie zoals invloedscoëfficiënten¹, stochasten en correlaties) wegschrijven. Eén JSON-bestand correspondeert met één fragility curve.

Om het converteren van de JSON-bestanden naar Hydra-Ring invoer (SQL-bestand) te ondersteunen is een Python script ontwikkeld (zie bijgevoegde map “ConvertDStabilitytoSQL”). Het script kan worden aangeroepen vanaf de commando-regel, maar ook vanuit een gebruikers grafische omgeving (GUI). Dit wordt hieronder toegelicht.

3.1.1 Eisen aan de D-Stability uitvoer

Gegeven valide JSON-bestanden zorgt het Python script voor een valide SQL-bestand. Hierbij wordt benadrukt dat de JSON-bestanden bij één dijkdoorsnede dienen te horen, en dat de opbouw van de dijk per bouwfase niet mag verschillen².

Er is één toegevoegde regel die specifiek op het Python script betrekking heeft: de labels van variabelen gekoppeld aan het parametertype “ModelFactor” dienen verschillend (uniek) te zijn, om te garanderen dat alle modelfactoren in de D-Stability uitvoer als aparte stochasten in de Hydra-Ring invoer voorkomen. De modelfactoren worden door D-Stability gelabeld met de naam van de bouwfase (stage) waar ze bij horen.

Om te zorgen dat de conversie van de JSON-bestanden naar het SQL-bestand zonder problemen verloopt, wordt het sterk aangeraden om in D-Stability unieke labels voor materiaallagen, states en bouwfases te gebruiken.

Ten aanzien van de Hydra-Ring invoer geldt er een aantal restricties die door Hydra-Ring zelf gevalideerd worden. Bijv. dienen de waterstanden in een fragility curve te stijgen, of moet het overslagdebiet positief zijn.

¹ Ofwel alfa-waarden.

² Het gaat om een beoordeling van een bestaand dijkvak.

3.1.2 De conversie aanroepen vanaf de commando-regel

Vanaf de commando-regel (DOS- of Anaconda Prompt) kan het Python script `dstabfragility_cli.py` voor conversie als volgt worden aangeroepen:

```
python dstabfragility_cli.py -i Invoerlijst -o Uitvoerbestand
```

`Invoerlijst` verwijst naar een tekstbestand (*.txt) waarin op elke regel het pad van een D-Stability uitvoerbestand (JSON) staat, elk met een overslagdebiet in $m^3/s/m$, gescheiden door een punt-komma, bijvoorbeeld:

```
Q:/data/ConvertDStabilitytoSQL/curve1.json;0.001200  
Q:/data/ConvertDStabilitytoSQL/curve2.json;0.100000
```

De waarde van het overslagdebiet dient groter dan nul te zijn. Ook in het geval van één fragility curve (één JSON-bestand) is het nodig om het overslagdebiet te definiëren. Het overslagdebiet wordt dan echter niet in Hydra-Ring berekeningen gebruikt.

`Uitvoerbestand` verwijst naar het SQL-bestand, waarin het resultaat van het conversie script opgenomen wordt.

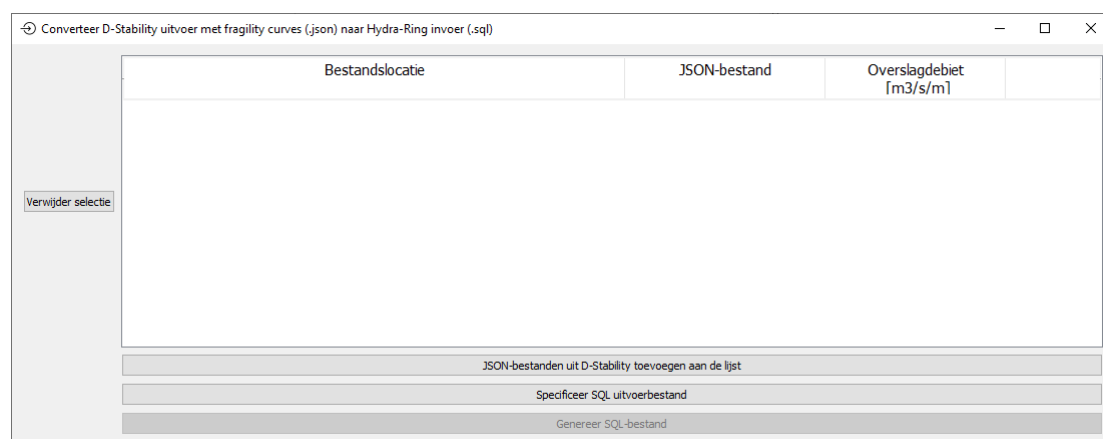
Eventueel kan `dstabfragility_cli.py` ook vanuit een geïntegreerde omgeving zoals Spyder worden gestart. In het geval van Spyder kan het beste vanuit de menu-optie “Run > Configuration per file...”. In het bovenste blok moet dan de optie “Execute in current console” geselecteerd zijn. Onder “General settings”, na het aanvinken van “Command line options”, dienen de opties

```
-i Invoerlijst -o Uitvoerbestand
```

te worden ingevuld. Het SQL-bestand wordt dan gegenereerd na het drukken op de knop “Run”.

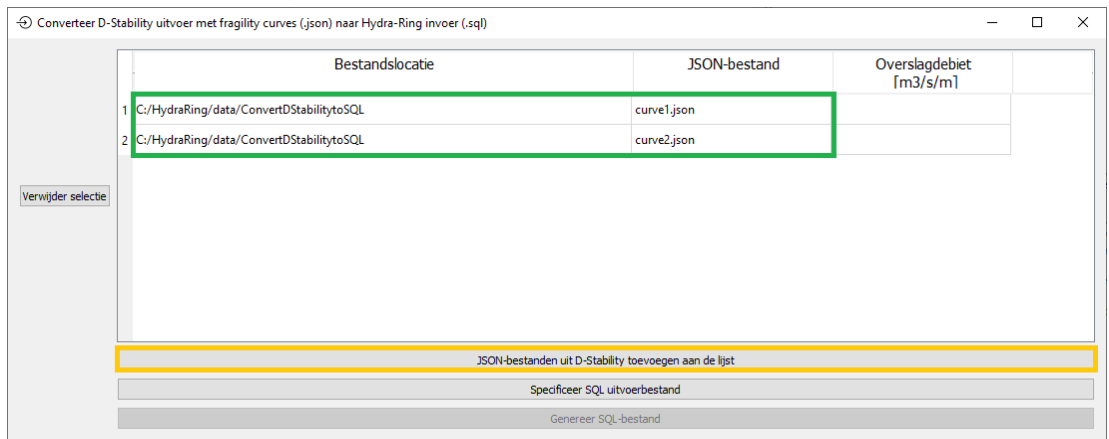
3.1.3 De conversie aanroepen vanuit de gebruikers grafische omgeving (GUI)

Het script `dstabfragility_gui.py` biedt een eenvoudige grafische schil aan en die kan vanuit een geïntegreerde omgeving (IDE) zoals Spyder worden opgestart³.

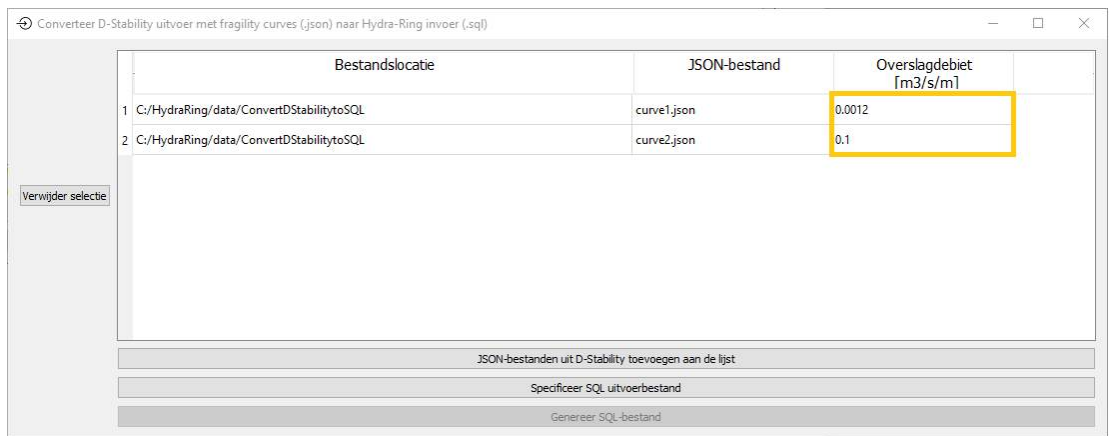


³ Om gebruik te maken van de GUI dient de Python module `PyQt5` vooraf te worden geïnstalleerd.

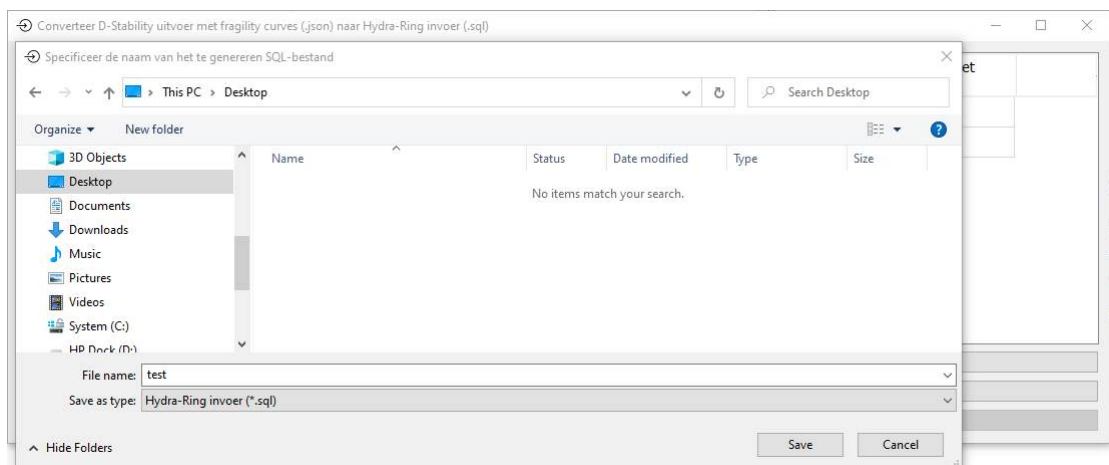
De bediening ervan is als volgt: de knop “*JSON-bestanden uit D-Stability toevoegen aan de lijst*” opent een dialoogvenster waarin JSON-bestanden kunnen worden geselecteerd (ook is het mogelijk om meerdere bestanden tegelijk te selecteren).



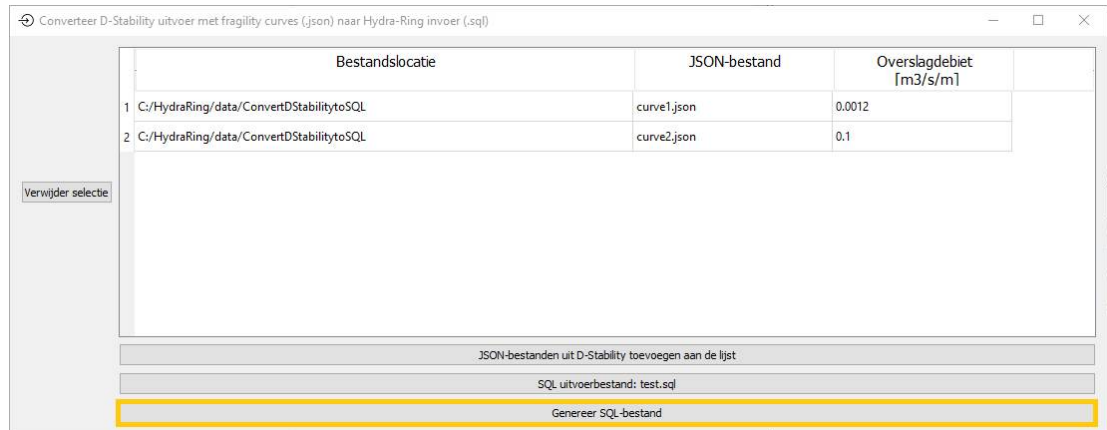
In kolom “*Overslagdebiet*” dient per bestand de juiste waarde van het overslagdebiet te worden ingevuld.



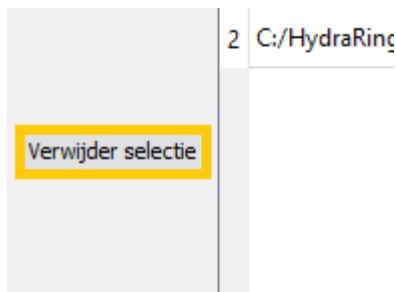
Met de knop “*Selecteer SQL uitvoerb bestand*” wordt het uitvoerb bestand gedefinieerd. Indien er een bestaand SQL-file geselecteerd wordt, dan wordt de inhoud van het bestand in de volgende stap overgeschreven.



De conversie wordt uitgevoerd door op de knop “*Genereer SQL-bestand*” te drukken. Deze knop is pas actief, als het uitvoerbestand bekend is.



Met de knop “*Verwijder selectie*” kunnen de geselecteerde SQL-bestanden uit de lijst worden weggehaald.



3.1.4 Uitvoer van het conversie script

De uitvoer van het conversie script bestaat uit een SQL-bestand (opgeslagen op de door de gebruiker geselecteerde locatie) en een CSV-bestand, *variables.csv* (standaard weggeschreven in de directory van het Python script).

Het SQL-bestand bevat de volgende tabellen (ofwel SQL-regels):

- **FragilityCurveVariablesPrefix**: groep identificaties van de sterktevariabelen uit D-Stability. De volgende groepen worden onderscheiden:
 - cohesion (*coh*)
 - friction angle (*phi*)
 - dilatancy angle (*dil*)
 - shear strength ratio (*shr*)
 - strength increase exponent (*str*)
 - POP (*pop*)
 - model factor (*mdf*)
 - SU table multiplier (*sux*)
- **FragilityCurveDischarge**: met overslagdebiet per fragility curve.
- **FragilityCurveVariables**: sterktevariabelen uit D-Stability en hun eigenschappen.
- **FragilityCurveCorrelations**: correlaties tussen de sterktevariabelen uit D-Stability.
- **FragilityCurveWaterlevelBeta**: waterstand en betrouwbaarheidsindex voor elk punt in een fragility curve.
- **FragilityCurveAlpha**: vector van invloedscoëfficiënten (alfa-waarden) voor elk punt in een fragility curve. Voor elke combinatie waterstand-betrouwbaarheidsindex is er één vector van alfa-waarden ter lengte van het totaal aantal variabelen.

De gegenereerde SQL-regels omvatten altijd eerst het leegmaken en erop volgend het invullen van de tabellen, niet de definitie zelf. Op deze manier wordt er dus niet aan een bestaande tabel toegevoegd.

Het bestand *variables.csv* (met “,” als de scheidingstekens) geeft een link tussen de sterktestochasten uit de JSON-bestanden en het SQL-bestand. De eerste kolom presenteert de sterktestochasten uit de JSON-bestanden, op de tweede kolom staan de bijbehorende stochasten uit het SQL-bestand. De namen van de sterktestochasten uit D-Stability zijn opgebouwd uit de identificatiegroep en de label gebruikt in D-Stability. De namen van de sterkte stochasten in het SQL-bestand zijn opgebouwd uit de identificatiegroep en een index.

	A	B
1	pop_Label 0	pop_1
2	shr_Label 1	shr_1
3	mdf_Model factor2_1	mdf_1
4	mdf_Model factor2_2	mdf_2
5	mdf_Model factor3_2	mdf_3
6	mdf_Model factor4_2	mdf_4
7	mdf_Model factor5_2	mdf_5

3.2 Klaarzetten SQL-bestand

Het SQL-bestand uit de vorige paragraaf dient nog met een aantal tabellen (SQL-regels) te worden uitgebreid. De belangrijkste daarvan zijn:

- *HydraulicModels*
- *Sections*
- *DesignTables*
- *Numerics*
- *VariableDatas*
- *CalculationProfiles*
- *SectionFaultTreeModels*
- *SectionSubMechanismModels*
- *ForelandModels*
- *Forelands*
- *Breakwaters*

Hieronder volgt een beschrijving van deze tabellen. In sommige dienen de invoervelden door de gebruiker te worden aangepast. Bijlage A geeft een voorbeeld van een volledig SQL-bestand weer. Alle tabellen zijn in detail in het Technisch Ontwerp van Hydra-Ring beschreven (Deltares, 2021a).

3.2.1 HydraulicModels

```
DELETE FROM [HydraulicModels];  
INSERT INTO [HydraulicModels] VALUES (1, 1, 'WTI 2017');
```

In deze tabel kan de tijdsintegratiemethode (met **kleur** hierboven gemarkeerd) aangepast worden. Hydra-Ring ondersteunt twee tijdsintegratiemethoden:

- FBC: id = 1
- NTI: id = 3

De overige invoervelden zijn standaard voor BOI en dienen niet aangepast te worden. De gekozen tijdsintegratiemethode heeft effect op de rekentijd en de nauwkeurigheid van de

resultaten. In het algemeen is de NTI methode nauwkeuriger dan FBC, maar neemt meer rekentijd in beslag. Beide methoden zijn beschreven in de Technische Handleiding van Hydra-Ring (Deltares, 2021b).

3.2.2 Sections

```
DELETE FROM [Sections];
INSERT INTO [Sections] VALUES (1, 1, 102, 'Lobith_test1', 'Lobith_test1', 0, 0, 0, 0,
14811048, 14811048, 100, 225, 0);
```

In deze tabel kunnen enkele invoervelden aangepast worden (met kleur gemarkeerd).

- De blauw gemarkeerde invoervelden (bijv. 'Lobith_test1') geven de naam van de berekening weer. Deze naam wordt gebruikt bij het genereren van de uitvoer bestanden.
- De geel gemarkeerde velden (bijv. 14811048) geven de id van de beschouwde hydraulische locatie op. De id correspondeert met de *LocationId* van de beschouwde locatie en kan in Riskeer of in de tabel *Locations* van het HLCD bestand opgezocht worden (dat wordt in Bijlage B toegelicht).
- Het groene getal (bijv. 225) staat voor de dijkoriëntatie in graden t.o.v. het Noorden. De dijkoriëntatie is nodig voor een berekening met meerdere fragility curves. In het geval van een berekening met één fragility curve zal de dijkoriëntatie genegeerd worden (en het kan dus een willekeurig getal tussen 0 en 360 graden ingevuld worden).
- Het paarse getal (bijv. 0) staat voor de lengte van het beschouwde vak in meters. In het geval van een berekening voor een dijkdoorsnede is de lengte gelijk aan 0 m.

De overige invoervelden zijn standaard voor BOI en dienen niet aangepast te worden.

3.2.3 DesignTables

```
DELETE FROM [DesignTables];
INSERT INTO [DesignTables] VALUES (1, 102, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0);
```

Deze tabel geeft aan dat het om een faalkansberekening⁴ gaat. Deze tabel dient niet aangepast te worden.

3.2.4 Numerics

```
DELETE FROM [Numerics];
INSERT INTO [Numerics] VALUES (1, 102, 1, 1, 201, 11, 4, 150, 0.15000001, 0.005, 0.005,
0.005, 2, 6, 10000, 40000, 0.1, -6, 6, 25);
```

In deze tabel kunnen de rekeninstellingen (met kleur gemarkeerd) aangepast worden. Het blauw gemarkeerde invoerveld geeft de type van de rekentechniek weer. De volgende rekentechnieken worden in Hydra-Ring onderscheiden:

- FORM: id = 1
- FORM met Directional Sampling: id = 11
- Directional Sampling met FORM voor het ontwerp punt: id = 12

⁴ Naast faalkansberekeningen kan Hydra-Ring nog ontwerp- en tabelberekeningen uitvoeren. Deze zijn voor het faalmechanisme macrostabiliteit minder relevant.

De geel gemarkeerde invoervelden geven respectievelijk de onder- en bovengrens van het aantal monsters gebruikt in de Directional Sampling methode. De overige invoervelden zijn standaard voor BOI en dienen niet aangepast te worden.

De gekozen rekentechniek heeft effect op de rekentijd en de nauwkeurigheid van de resultaten. In het algemeen is methode 12 nauwkeuriger dan methode 11, en deze is zelf ook nauwkeuriger dan methode 1. Methode 12 neemt de langste rekentijd in beslag; en methode 1 de kortste. Het wordt aanbevolen om methode 11 of 12 te gebruiken. De praktijkervaring leert dat methode 1 vaak tot onbetrouwbare resultaten leidt. De rekentechnieken zijn beschreven in het functioneel ontwerp van de Probabilistische bibliotheek van BOI (Deltares, 2021c).

3.2.5 VariableDatas

```
DELETE FROM [VariableDatas];
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 1, 16.90, 0, 0, NULL, NULL, NULL, 1, 0, 300);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 10, 0, 19, 4.75, 0.5, 0, 99, 1, 0, 300);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 11, 0, 19, 2.6, 0.35, 0, 99, 1, 0, 300);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 12, 1, 0, 0, NULL, NULL, NULL, 1, 0, 300);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 120, 0, 19, 1, 0.07, 0, 99, 1, 0, 300);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 123, 0, 19, 0.92, 0.24, 0, 99, 1, 0, 300);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 30, 0, 2, 0, 1, NULL, NULL, 1, 0, 999999);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 31, 0, 2, 0, 1, NULL, NULL, 1, 0, 999999);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 32, 0, 2, 0, 1, NULL, NULL, 1, 0, 999999);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 33, 0, 2, 0, 1, NULL, NULL, 1, 0, 999999);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 34, 0, 2, 0, 1, NULL, NULL, 1, 0, 999999);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 35, 0, 2, 0, 1, NULL, NULL, 1, 0, 999999);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 36, 0, 2, 0, 1, NULL, NULL, 1, 0, 999999);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 37, 0, 2, 0, 1, NULL, NULL, 1, 0, 999999);
```

Deze tabel geeft informatie over de kansverdelingen van de gebruikte sterkte stochasten. De id's van de stochasten zijn in de bovenstaande tabel vetgedrukt:

- Stochasten met id 1, 10, 11, 12, 120 en 123 worden gebruikt in de overslagberekeningen.
- Stochasten met id 30 t/m 37 zijn een onderdeel van de Z-functie. Dit zijn speciale stochasten, die altijd standaard normaal verdeeld zijn. Deze stochasten zijn een hulpmiddel bij het vertalen van de informatie uit fragility curves naar de Z-functie.

In deze tabel kan de dijkhoogte (blauw gemarkeerd) aangepast worden. De dijkhoogte, in meters + NAP, is in dit geval een deterministische variabele. De overige invoervelden zijn standaard voor BOI en dienen niet aangepast te worden.

3.2.6 CalculationProfiles

```
DELETE FROM [CalculationProfiles];
INSERT INTO [CalculationProfiles] VALUES (1, 1, 0.000, 10.000, 1.0);
INSERT INTO [CalculationProfiles] VALUES (1, 2, 31.480, 16.00, 1.0);
```

In deze tabel wordt de dijkprofiel gedefinieerd ten behoeve van de overslagberekeningen. Het gaat in deze tabel altijd om een dijkprofiel zonder voorland (ook in het geval van berekeningen met één fragility curve dient deze tabel te worden ingevuld). De geel gemarkeerde getallen geven de x-coördinaten van een profiel in meters weer. De blauw gemarkeerde getallen staan voor de z-coördinaten van een profiel in meters + NAP. De laatste getallen geven de ruwheidcoëfficiënt weer.

3.2.7 SectionFaultTreeModels

```
DELETE FROM [SectionFaultTreeModels];  
INSERT INTO [SectionFaultTreeModels] VALUES (1, 102, 1, 1, 2001);
```

Deze tabel bevat de id van het gebruikte foutenboom (2001). Voor het faalmechanisme bestaat het foutenboom uit slechts een sub-mechanisme. Deze tabel dient niet aangepast te worden.

3.2.8 SectionSubMechanismModels

```
DELETE FROM [SectionSubMechanismModels];  
INSERT INTO [SectionSubMechanismModels] VALUES ( 1 , 1 , 1 , 201 , 96);
```

Deze tabel bevat de id van de gebruikte interpolatietype. Het betreft de interpolatie over het overslaggebied in het geval van meerdere fragility curves (dus tussen verschillende fragility curves bij een gegeven waterstand). De volgende interpolatietypen worden in Hydra-Ring onderscheiden voor het faalmechanisme macrostabiliteit binnenwaarts:

- Log-lineaire interpolatie: id = 96 (default)
- Lineaire interpolatie: id = 97

De log-lineaire interpolatie is conservatiever dan de lineaire interpolatie, en daarvoor wordt deze als default in Hydra-Ring gebruikt.

3.2.9 ForelandModels

```
DELETE FROM [ForelandModels];  
INSERT INTO [ForelandModels] VALUES (1, 102, 3);
```

Deze tabel is alleen van toepassing op een berekening met een dam en/of voorland. De inhoud van deze tabel dient niet aangepast te worden.

In het geval van berekeningen zonder dam en/of voorland dient alleen

DELETE FROM [ForelandModels]; in het SQL-bestand te worden opgenomen.

3.2.10 Forelands

```
DELETE FROM [Forelands];  
INSERT INTO [Forelands] VALUES (1, 1, -50, 9);  
INSERT INTO [Forelands] VALUES (1, 2, 0, 10);
```

Deze tabel is alleen van toepassing op een berekening met een voorland. De geel aangeduide getallen geven de x-coördinaten van het voorlandprofiel in meters weer. De blauw opgelichte getallen geven de z-coördinaten in meters + NAP weer.

In het geval van berekeningen zonder dam en/of voorland dient alleen

DELETE FROM [Forelands]; in het SQL-bestand te worden opgenomen.

3.2.11 Breakwaters

```
DELETE FROM [Breakwaters];  
INSERT INTO [Breakwaters] VALUES (1, 1, 2.0);
```

Deze tabel is alleen van toepassing op een berekening met een dam. Het geel gemarkeerde getal geeft een damtype weer. Er worden drie damtypes in Hydra-Ring onderscheiden:

- Caisson: id = 1
- Muur: id = 2
- Havendam: id = 3

De damhoogte, in meters + NAP, wordt in het blauw aangeduide veld weergegeven.

In het geval van berekeningen zonder dam en/of voorland dient alleen

DELETE FROM [Breakwaters]; in het SQL-bestand te worden opgenomen.

4 Inspectie van Hydra-Ring resultaten

De resultaten van de berekeningen worden weggeschreven naar het bestand **-output.sqlite* (te vinden in de map "Resultaten"). Het bestand kan bijv. met het programma "DB Browser for SQLite" worden geïnspecteerd. Deze applicatie kan gratis van het internet worden gedownload.

Het eindantwoord, in de vorm van een betrouwbaarheidsindex, staat vermeld in de tabel *DesignBeta* (kolom *BetaValue*) na het selecteren van *LevelTypeId* = 4. Het omrekenen van betrouwbaarheidsindex naar faalkans (en vice versa) wordt in Bijlage C beschreven.

Table: DesignBeta

OuterIterationId	SectionId	LayerId	Alternatvleid	FaultTreeId	LevelTypeId	SubMechanismId	PeriodId	WindDirectionId	ClosingSituationId	BetaValue
Filter	Filter	Filter	Filter	Filter	=4	Filter	Filter	Filter	Filter	Filter
1	1	1	1	-999	4	201	3	0	0	2.99538

Voorbeeld eindantwoord

De betrouwbaarheidsindex per windrichting kan ook uit de tabel worden gelezen na het selecteren van *LevelTypeId* = 7 en *ClosingSituationId* = 1 (en/of 2 in systemen met een stormvloedkering⁵). In het geval van de tijdsintegratiemethode FBC is *PeriodId* altijd gelijk aan 3. Bij de tijdsintegratiemethode NTI dient aanvullend *PeriodId* = 1 te worden geselecteerd.

Table: DesignBeta

OuterIterationId	SectionId	LayerId	Alternatvleid	FaultTreeId	LevelTypeId	SubMechanismId	PeriodId	WindDirectionId	ClosingSituationId	BetaValue
Filter	Filter	Filter	Filter	Filter	=7	Filter	Filter	Filter	=1	Filter
1	1	1	1	-999	7	201	3	1	1	4.87769
2	1	1	1	-999	7	201	3	2	1	4.87769
3	1	1	1	-999	7	201	3	3	1	4.87769
4	1	1	1	-999	7	201	3	4	1	4.87769
5	1	1	1	-999	7	201	3	5	1	4.87769
6	1	1	1	-999	7	201	3	6	1	4.87769
7	1	1	1	-999	7	201	3	7	1	4.87769
8	1	1	1	-999	7	201	3	8	1	4.86696
9	1	1	1	-999	7	201	3	9	1	4.86547

Voorbeeld betrouwbaarheidsindex per windrichting

De id van de maatgevende windrichting staat vermeld in de tabel *GoverningWind* (kolom *WindDirectionId*). In het geval van de tijdsintegratiemethode NTI dient aanvullend *PeriodId* = 1 te worden geselecteerd.

Table: GoverningWind

OuterIterationId	PeriodId	WindDirectionId
Filter	Filter	Filter
1	3	8

Voorbeeld id van de maatgevende windrichting

Op basis van de gevonden *WindDirectionId* kan vervolgens de maatgevende windrichting uit de tabel *WindDirections* worden gelezen.

⁵ Europortkering of Ramspolkering.

WindDirectionId	WindDirectionAngle	WindDirectionName
Filter	Filter	Filter
1	0.0	N
2	22.5	NNO
3	45.0	NO
4	67.5	ONO
5	90.0	O
6	112.5	OZO
7	135.0	ZO
8	157.5	ZZO
9	180.0	Z
10	202.5	ZZW
11	225.0	ZW
12	247.5	WZW
13	270.0	W
14	292.5	WNW
15	315.0	NW
16	337.5	NNW

Voorbeeld maatgevende windrichting

Gegeven de id van de maatgevende windrichting kan het hoofdillustratiepunt worden uitgelezen. Het hoofdillustratiepunt is verspreid over drie tabellen:

- *DesignAlpha*
- *DesignAlphaFragilityCurveVariables*
- *DesignPointResults*

Het hoofdillustratiepunt voor de overslag-gerelateerde stochasten en de basisstochasten (zoals rivierafvoer, zeewaterstand, meerpeil, windsnelheid, modelonzekerheden en statistische onzekerheden) staat vermeld in de tabel *DesignAlpha* na het selecteren van *LevelTypeId = 7*, *WindDirectionId* van de maatgevende windrichting (8 in dit voorbeeld), *ClosingSituationId = 1* (en/of 2 in systemen met een stormvloedkering). In het geval van de tijdsintegratiemethode NTI dient aanvullend *PeriodId = 1* te worden geselecteerd.

Het illustratiepunt staat dan vermeld in kolommen *StochastId*, *AlphaValue*, *Duration* en *X*. Kolom *StochastId* geeft de id van een stochast. De naam van de stochast kan in de tabel *Stochasts* worden opgezocht. Kolom *AlphaValue* geeft de waarde van de invloedscoëfficiënt van een stochast, kolom *Duration* bevat de duur van een stochast in uren en in kolom *X* is de waarde van een stochast in het ontwerppunt terug te vinden. Stochasten met ids 30 t/m 37 zijn hulpstochasten in Hydra-Ring en vallen hier buiten de interesse om.

OuterIterationId	SectionId	LayerId	AlternatId	FaultTreeId	LevelTypeId	SubMechanismId	PeriodId	WindDirectionId	ClosingSituationId	StochastId	AlphaValue	Duration	X
Filter	Filter	Filter	Filter	Filter	=7	Filter	Filter	=8	=1	Filter	Filter	Filter	Filter
1	1	1	1	-999	7	201	3	8	1	1	0.0	4383.0	16.9
2	1	1	1	-999	7	201	3	8	1	10	0.0	4383.0	4.75
3	1	1	1	-999	7	201	3	8	1	11	0.0	4383.0	2.6
4	1	1	1	-999	7	201	3	8	1	12	0.0	4383.0	1.0
5	1	1	1	-999	7	201	3	8	1	30	0.0251895	4383.0	-0.113406
6	1	1	1	-999	7	201	3	8	1	31	0.0503789	4383.0	-0.226812
7	1	1	1	-999	7	201	3	8	1	32	0.0755684	4383.0	-0.340219
8	1	1	1	-999	7	201	3	8	1	33	0.100758	4383.0	-0.453625
9	1	1	1	-999	7	201	3	8	1	34	0.125947	4383.0	-0.567031
10	1	1	1	-999	7	201	3	8	1	35	0.151137	4383.0	-0.680437
11	1	1	1	-999	7	201	3	8	1	36	0.176326	4383.0	-0.793844
12	1	1	1	-999	7	201	3	8	1	37	0.251895	4383.0	-1.13406
13	1	1	1	-999	7	201	3	8	1	120	0.0	4383.0	1.0
14	1	1	1	-999	7	201	3	8	1	123	0.0	4383.0	0.920019
15	1	1	1	-999	7	201	3	8	1	141	0.0	12.0	0.0
16	1	1	1	-999	7	201	3	8	1	143	-0.905924	12.212	13139.6
17	1	1	1	-999	7	201	3	8	1	199	0.0	12.0	5.60651
18	1	1	1	-999	7	201	3	8	1	243	-0.158038	12.212	348.721
19	1	1	1	-999	7	201	3	8	1	299	0.0	12.0	1.00001
20	1	1	1	-999	7	201	3	8	1	340	-0.0452053	4383.0	0.0305281
21	1	1	1	-999	7	201	3	8	1	341	0.0	4383.0	0.96
22	1	1	1	-999	7	201	3	8	1	342	0.0	4383.0	1.03

Voorbeeld hoofdillustratiepunt voor de overslag-gerelateerde stochasten en de basisstochasten

Het hoofdillustratiepunt voor de stochasten uit D-Stability staat vermeld in de tabel *DesignAlphaFragilityCurveVariables* na het selecteren van *LevelTypeId* = 7, *WindDirectionId* van de maatgevende windrichting (8 in dit voorbeeld), *ClosingSituationId* = 1 (en/of 2 in systemen met een stormvloedkering). In het geval van de tijdsintegratiemethode NTI dient aanvullend *PeriodId* = 1 te worden geselecteerd.

Het illustratiepunt staat dan vermeld in kolommen *FragilityCurveVariableId*, *AlphaValue*, *Duration* en *X*. Kolom *FragilityCurveVariableId* geeft de id van een stochast, de naam van de stochast kan in de tabel *FragilityCurveVariables* worden opgezocht.

OuterIterationId	SectionId	LayerId	AlternativeId	LevelTypeId	SubMechanismId	PeriodId	WindDirectionId	ClosingSituationId	FragilityCurveVariableId	AlphaValue	Duration	X
1	1	1	1	7	201	3	8	1	1	0.0251895	4383.0	9.82692
2	1	1	1	7	201	3	8	1	2	0.0551128	4383.0	10.0
3	1	1	1	7	201	3	8	1	3	0.0764568	4383.0	-0.340219
4	1	1	1	7	201	3	8	1	4	0.0551128	4383.0	-0.453625
5	1	1	1	7	201	3	8	1	5	0.146647	4383.0	-0.567031
6	1	1	1	7	201	3	8	1	6	0.151137	4383.0	-0.680437
7	1	1	1	7	201	3	8	1	7	0.221352	4383.0	-0.793844
8	1	1	1	7	201	3	8	1	8	0.299873	4383.0	-1.13406

Voorbeeld hoofdillustratiepunt voor de stochasten uit D-Stability

Het hoofdillustratiepunt voor de hydraulische belastingen (zoals waterstand of golfhoogte) en de waarde van de Z-functie staan vermeld in de tabel *DesignPointResults* na het selecteren van *WindDirectionId* van de maatgevende windrichting (8 in dit voorbeeld) en *ClosingSituationId* = 1 (en/of 2 in systemen met een stormvloedkering). In het geval van de tijdsintegratiemethode NTI dient aanvullend *PeriodId* = 1 te worden geselecteerd.

Het illustratiepunt staat vermeld in kolommen *OutputVariableId* en *Value*. Kolom *OutputVariableId* geeft de id van een stochast, de naam van de stochast kan in de tabel *OutputVariables* worden teruggevonden. Kolom *Value* geeft de waarde van een stochast in het ontwerppunt weer.

OuterIterationId	SectionId	LayerId	AlternativeId	SubMechanismId	PeriodId	WindDirectionId	ClosingSituationId	OutputVariableId	Value	IsInterpolated
1	1	1	1	201	3	8	1	1	0.00014912	NULL
2	1	1	1	201	3	8	1	2	16.8645	1
3	1	1	1	201	3	8	1	3	0.210473	1
4	1	1	1	201	3	8	1	7	1.75219	1
5	1	1	1	201	3	8	1	8	157.5	1
6	1	1	1	201	3	8	1	412	0.0168692	NULL

Voorbeeld hoofdillustratiepunt voor de hydraulische belastingen

De illustratiepunten voor de windrichtingen anders dan de maatgevende windrichting kunnen op analoge wijze opgezocht worden door de waarde van *WindDirectionId* aan te passen.

5 Referenties

Deltares, 2021a. *Technical Design of Hydra-Ring 20.1.3*. Deltares, 11206818-015-0002, februari 2021.

Deltares, 2021b. *Technical Reference Manual of Hydra-Ring 20.1.3*. Deltares, 11206818-015-GEO-0003, februari 2021.

Deltares, 2021c. *Functional Design of the Probabilistic Library 20.1.1*. Deltares, 11206818-015-0006, februari 2021.

Spee, 2020. *Macrostability Inwards in Hydra-Ring: technical details*. Deltares, 11205758-020-GEO-0002, september 2020.

A Voorbeeld van SQL-bestand

```
DELETE FROM [FragilityCurveVariablesPrefix];
INSERT INTO [FragilityCurveVariablesPrefix] VALUES(102, 30, 'coh');
INSERT INTO [FragilityCurveVariablesPrefix] VALUES(102, 31, 'phi');
INSERT INTO [FragilityCurveVariablesPrefix] VALUES(102, 32, 'dil');
INSERT INTO [FragilityCurveVariablesPrefix] VALUES(102, 33, 'shr');
INSERT INTO [FragilityCurveVariablesPrefix] VALUES(102, 34, 'str');
INSERT INTO [FragilityCurveVariablesPrefix] VALUES(102, 35, 'pop');
INSERT INTO [FragilityCurveVariablesPrefix] VALUES(102, 36, 'mdf');
INSERT INTO [FragilityCurveVariablesPrefix] VALUES(102, 37, 'sux');

DELETE FROM [FragilityCurveDischarge];
INSERT INTO [FragilityCurveDischarge] VALUES (1, 102, 1, 1, 1, 0.001);
INSERT INTO [FragilityCurveDischarge] VALUES (1, 102, 1, 1, 2, 0.01);

DELETE FROM [FragilityCurveVariables];
INSERT INTO [FragilityCurveVariables] VALUES (1, 102, 1, 1, 'coh', 4, 10, 1, 2, NULL);
INSERT INTO [FragilityCurveVariables] VALUES (1, 102, 1, 1, 'phi', 0, 10, 0, NULL, NULL);
INSERT INTO [FragilityCurveVariables] VALUES (1, 102, 1, 1, 'dil', 2, 0, 1, NULL, NULL);
INSERT INTO [FragilityCurveVariables] VALUES (1, 102, 1, 1, 'shr', 2, 0, 1, NULL, NULL);
INSERT INTO [FragilityCurveVariables] VALUES (1, 102, 1, 1, 'str', 2, 0, 1, NULL, NULL);
INSERT INTO [FragilityCurveVariables] VALUES (1, 102, 1, 1, 'pop', 2, 0, 1, NULL, NULL);
INSERT INTO [FragilityCurveVariables] VALUES (1, 102, 1, 1, 'mdf', 2, 0, 1, NULL, NULL);
INSERT INTO [FragilityCurveVariables] VALUES (1, 102, 1, 1, 'sux', 2, 0, 1, NULL, NULL);

DELETE FROM [FragilityCurveCorrelations];
INSERT INTO [FragilityCurveCorrelations] VALUES (1, 102, 1, 1, 'coh', 'phi', 0.253);
INSERT INTO [FragilityCurveCorrelations] VALUES (1, 102, 1, 1, 'coh', 'dil', 0.15);
INSERT INTO [FragilityCurveCorrelations] VALUES (1, 102, 1, 1, 'coh', 'shr', 0.65);
INSERT INTO [FragilityCurveCorrelations] VALUES (1, 102, 1, 1, 'phi', 'shr', 1.0);
INSERT INTO [FragilityCurveCorrelations] VALUES (1, 102, 1, 1, 'dil', 'str', 0.555);
INSERT INTO [FragilityCurveCorrelations] VALUES (1, 102, 1, 1, 'shr', 'mdf', 0.99);
INSERT INTO [FragilityCurveCorrelations] VALUES (1, 102, 1, 1, 'shr', 'sux', -0.6);
INSERT INTO [FragilityCurveCorrelations] VALUES (1, 102, 1, 1, 'pop', 'mdf', 0.85);
INSERT INTO [FragilityCurveCorrelations] VALUES (1, 102, 1, 1, 'mdf', 'sux', 0.39);

DELETE FROM [FragilityCurveWaterLevelBeta];
INSERT INTO [FragilityCurveWaterLevelBeta] VALUES (1, 102, 1, 1, 1, 1, 12.0, 10.0);
INSERT INTO [FragilityCurveWaterLevelBeta] VALUES (1, 102, 1, 1, 1, 2, 15.0, 5.0);
INSERT INTO [FragilityCurveWaterLevelBeta] VALUES (1, 102, 1, 1, 2, 1, 12.0, 5.0);
INSERT INTO [FragilityCurveWaterLevelBeta] VALUES (1, 102, 1, 1, 2, 2, 15.0, 3.0);

DELETE FROM [FragilityCurveAlpha];
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 1, 1, 'coh', 0.1);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 1, 1, 'phi', 0.2);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 1, 1, 'dil', 0.3);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 1, 1, 'shr', 0.4);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 1, 1, 'str', 0.5);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 1, 1, 'pop', 0.6);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 1, 1, 'mdf', 0.7);
```

```

INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 1, 1, 'sux', 1.0);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 1, 2, 'coh', 0.1);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 1, 2, 'phi', 0.2);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 1, 2, 'dil', 0.3);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 1, 2, 'shr', 0.4);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 1, 2, 'str', 0.5);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 1, 2, 'pop', 0.6);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 1, 2, 'mdf', 0.7);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 1, 2, 'sux', 1.0);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 2, 1, 'coh', 0.1);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 2, 1, 'phi', 0.2);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 2, 1, 'dil', 0.3);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 2, 1, 'shr', 0.4);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 2, 1, 'str', 0.5);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 2, 1, 'pop', 0.6);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 2, 1, 'mdf', 0.7);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 2, 1, 'sux', 1.0);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 2, 2, 'coh', 0.1);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 2, 2, 'phi', 0.2);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 2, 2, 'dil', 0.3);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 2, 2, 'shr', 0.4);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 2, 2, 'str', 0.5);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 2, 2, 'pop', 0.6);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 2, 2, 'mdf', 0.7);
INSERT INTO [FragilityCurveAlpha] VALUES (1, 102, 1, 1, 2, 2, 'sux', 1.0);

DELETE FROM [HydraulicModels];
INSERT INTO [HydraulicModels] VALUES (1, 1, 'WTI 2017');

DELETE FROM [Sections];
INSERT INTO [Sections] VALUES (1, 1, 102, 'Lobith_test1', 'Lobith_test1', 0, 0, 0, 0, 14811048,
14811048, 100, 225, 0);

DELETE FROM [DesignTables];
INSERT INTO [DesignTables] VALUES (1, 102, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0);

DELETE FROM [Numerics];
INSERT INTO [Numerics] VALUES (1, 102, 1, 1, 201, 11, 4, 150, 0.15000001, 0.005, 0.005, 0.005,
2, 6, 10000, 40000, 0.1, -6, 6, 25);

DELETE FROM [VariableDatas];
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 1, 16.90, 0, 0, NULL, NULL, NULL, 1, 0,
300);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 10, 0, 19, 4.75, 0.5, 0, 99, 1, 0, 300);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 11, 0, 19, 2.6, 0.35, 0, 99, 1, 0, 300);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 12, 1, 0, 0, NULL, NULL, NULL, 1, 0, 300);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 120, 0, 19, 1, 0.07, 0, 99, 1, 0, 300);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 123, 0, 19, 0.92, 0.24, 0, 99, 1, 0, 300);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 30, 0, 2, 0, 1, NULL, NULL, 1, 0, 999999);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 31, 0, 2, 0, 1, NULL, NULL, 1, 0, 999999);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 32, 0, 2, 0, 1, NULL, NULL, 1, 0, 999999);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 33, 0, 2, 0, 1, NULL, NULL, 1, 0, 999999);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 34, 0, 2, 0, 1, NULL, NULL, 1, 0, 999999);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 35, 0, 2, 0, 1, NULL, NULL, 1, 0, 999999);

```



```

INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 36, 0, 2, 0, 1, NULL, NULL, 1, 0, 999999);
INSERT INTO [VariableDatas] VALUES (1, 102, 1, 1, 37, 0, 2, 0, 1, NULL, NULL, 1, 0, 999999);

DELETE FROM [CalculationProfiles];
INSERT INTO [CalculationProfiles] VALUES ( 1, 1, 0.000, 10.000, 1.0);
INSERT INTO [CalculationProfiles] VALUES ( 1, 2, 31.480, 16.900, 1.0);

DELETE FROM [SectionFaultTreeModels];
INSERT INTO [SectionFaultTreeModels] VALUES (1, 102, 1, 1, 2001);

DELETE FROM [SectionSubMechanismModels];
INSERT INTO [SectionSubMechanismModels] VALUES ( 1 , 1 , 1 , 201 , 96);

DELETE FROM [ForelandModels];
INSERT INTO [ForelandModels] VALUES (1, 102, 3);

DELETE FROM [Forelands];
INSERT INTO [Forelands] VALUES (1, 1, -50, 9);
INSERT INTO [Forelands] VALUES (1, 2, 0, 10);

DELETE FROM [Breakwaters];
INSERT INTO [Breakwaters] VALUES (1, 1, 2);

DELETE FROM [PreprocessorSettings];
DELETE FROM [Fetches];
DELETE FROM [AreaPoints];
DELETE FROM [PresentationSections];
DELETE FROM [Profiles];
DELETE FROM [ProbabilityAlternatives];
DELETE FROM [SetUpHeights];
DELETE FROM [CalcWindDirections];
DELETE FROM [Swells];
DELETE FROM [WaveReductions];
DELETE FROM [Areas];
INSERT INTO [Areas] VALUES (1, '1', 'Nederland');
DELETE FROM [Projects];
INSERT INTO [Projects] VALUES (1, 'BOI', 'Hydra-Ring calculation');

```

B Opzoeken van LocationId

B.1 In Riskeer

De *LocationId* van een hydraulische locatie kan in Riskeer gevonden worden na het koppelen van de hydraulische database en openen van het overzicht van alle locaties (zie kolom *ID*).

	Berekenen	Illustratiepunten inlezen	Naam	ID	Coördinaten [m]	Waterstand [m+NAP]
▶	<input type="checkbox"/>	<input type="checkbox"/>	YM_2_12-2_dk_00012	700131	(132665, 547984)	-
	<input type="checkbox"/>	<input type="checkbox"/>	YM_1_12-2_dk_00022	700132	(133405, 546510)	-
	<input type="checkbox"/>	<input type="checkbox"/>	YM_2_12-2_dk_00035	700133	(134311, 545007)	-
	<input type="checkbox"/>	<input type="checkbox"/>	YM_1_12-2_dk_00051	700134	(135312, 542706)	-
	<input type="checkbox"/>	<input type="checkbox"/>	YM_2_12-2_dk_00125	700135	(136020, 532274)	-
	<input type="checkbox"/>	<input type="checkbox"/>	YM_1_12-2_dk_00106	700136	(136251, 534813)	-
	<input type="checkbox"/>	<input type="checkbox"/>	YM_1_12-2_dk_00089	700137	(136483, 537353)	-
	<input type="checkbox"/>	<input type="checkbox"/>	YM_2_12-2_dk_00072	700138	(136587, 540167)	-
	<input type="checkbox"/>	<input type="checkbox"/>	YM_2_12-2_dk_00076	700139	(136687, 539593)	-

B.2 Direct in database

HRD en HLCD bestanden zijn beide sqlite databases, die met applicatie “DB Browser for SQLite” geopend kunnen worden.

Open een HRD database (bijv. “WBI2017_IJsselmeer_12-2_v02.sqlite”) en ga naar de tabel *HRDLocations*. Deze tabel geeft een overzicht van alle hydraulische locaties in de database inclusief de namen, en de x- en y-coördinaten. Voor de gewenste locatie (bijv. YM_2_12-2_dk_00012) noteer de bijbehorende *HRDLocationId* (131 in dit geval).

	HRDLocationId	LocationTypeId	Name	XCoordinate	YCoordinate	WaterLevelCorrection
	Filter	Filter	Filter	Filter	Filter	Filter
1	131	4	YM_2_12-2_dk_00012	132665.0	547984.0	0.0
2	132	4	YM_1_12-2_dk_00022	133405.0	546510.0	0.0
3	133	4	YM_2_12-2_dk_00035	134311.0	545007.0	0.0
4	134	4	YM_1_12-2_dk_00051	135312.0	542706.0	0.0
5	135	4	YM_2_12-2_dk_00125	136020.0	532274.0	0.0
6	136	4	YM_1_12-2_dk_00106	136251.0	534813.0	0.0
7	137	4	YM_1_12-2_dk_00089	136483.0	537353.0	0.0
8	138	4	YM_2_12-2_dk_00072	136587.0	540167.0	0.0
9	139	4	YM_2_12-2_dk_00076	136687.0	539593.0	0.0

Open het bijbehorende HLCD bestand en ga naar de tabel *Tracks*. Zoek de naam van de HRD database in kolom *HRDFileName* op en noteer de bijbehorende *TrackId* (7059 in dit geval).

Table: **Tracks**

TrackId	RegionId	Name	HRDFileName
Filter	Filter	Filter	WBI2017_Dijsselmeer_12-2_v02
1 7059	7	12-2	WBI2017_Dijsselmeer_12-2_v02.sqlite

In het HLCD bestand ga naar de tabel *Locations*. Gegeven de waarden van *TrackId* en *HRDLocationId* zoek naar de bijbehorende waarde van *LocationId* (700131 in dit geval).

Table: **Locations**

LocationId	TypeOfHydraulicDataId	TrackId	HRDLocationId	Area
Filter	Filter	=7059	=131	Filter
1 700131	2	7059	131	NULL

C Omrekenen betrouwbaarheidsindex ↔ faalkans

C.1 Van betrouwbaarheidsindex naar faalkans

In Python

```
from scipy.stats import norm

betrouwbaarheidsindex = 4.0
faalkans = norm.cdf(-betrouwbaarheidsindex)
```

In Excel

	A	B	C	D	E
1	betrouwbaarheidsindex	4			
2	faalkans	=NORM.DIST(-B1;0;1;1)			
3		NORM.DIST(x; mean; standard_dev; cumulative)			
4					

C.2 Van faalkans naar betrouwbaarheidsindex

In Python

```
from scipy.stats import norm

faalkans = 0.001

betrouwbaarheidsindex = norm.isf(faalkans)
```

In Excel

	A	B	C	D	E
1	faalkans	0.001			
2	betrouwbaarheidsindex	=-NORM.INV(B1;0;1)			
3		NORM.INV(probability; mean; standard_dev)			
4					

Deltares is een onafhankelijk kennisinstituut voor toegepast onderzoek op het gebied van water en ondergrond. Wereldwijd werken we aan slimme oplossingen voor mens, milieu en maatschappij.

Deltares

www.deltares.nl