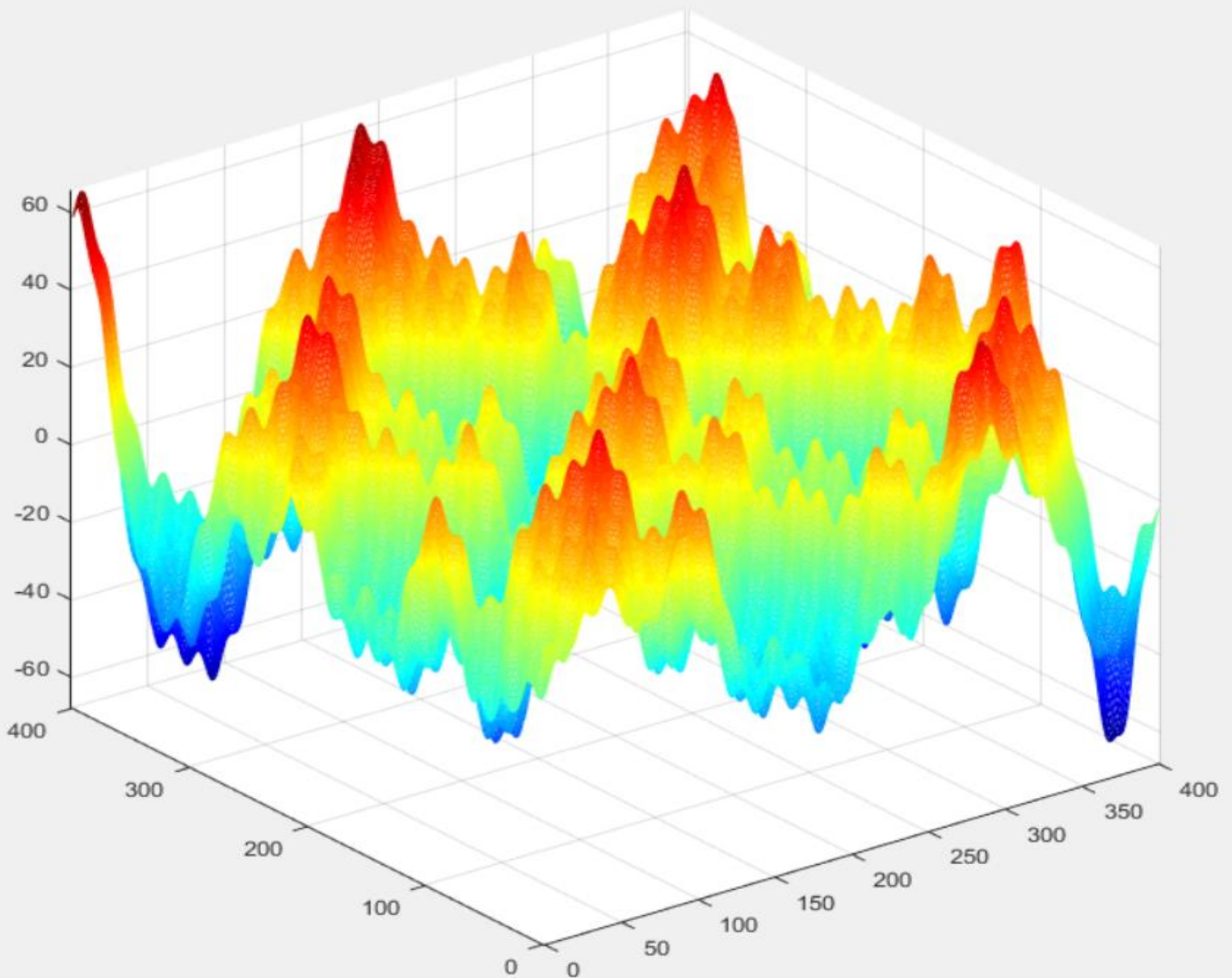


T. van de Velde

Extreme waarden van stochastische velden



Extreme waarden van stochastische velden

Door

T. van de Velde

BSc-eindwerk

Bachelor of Science
Civiele techniek
Technische Universiteit Delft

Studentnummer: 4301730
E-mail: t.velde94@gmail.com
Datum: 23-5-2016

1^e begeleider: Dr.ir. P.C.J. Hoogenboom
2^e begeleider: Dr.ir. M.A.N. Hendriks

Voorwoord

Dit rapport is opgesteld in het kader van het afsluiten van mijn bachelor Civiele Techniek aan de technische universiteit Delft als bachelor-eindwerk. Het onderwerp bouwt voort op een MSc-eindwerk uitgevoerd door B. Elferink waarin verschillende schaalconstructies nauwkeurig gemeten zijn. Voor dit onderzoek heb ik in Matlab een programma geschreven waarmee ik data kan genereren en verzamelen over extreme waarden van stochastische velden. Dit onderzoek is hierdoor vrij wiskundig van aard, maar het heeft als primair doel het verbeteren van de berekeningen op schaalconstructies.

Dit rapport is geschreven voor mensen met een achtergrond in de civiele techniek en hierdoor een basiskennis hebben van constructies, kansrekening en statistiek wiskunde.

Inhoud

Samenvatting	5
1 Inleiding.....	6
1.1 Probleemstelling	6
1.2 Doelstelling.....	6
1.3 Aanpak/leeswijzer.....	6
2 Achterliggende theorie	7
2.1 Schaalconstructies.....	7
2.2 Vormimperfecties	7
2.3 Modeleren van vormimperfecties	8
3 Creëren van stochastische velden	10
3.1 Controle van het geschreven programma.....	12
4 Verdeling van de extreme waarden.....	14
4.1 Gegenereerde data.....	14
4.2 Het vinden van de best fittende verdeling: Matlab	18
4.3 Het vinden van de best fittende verdeling: EasyFit	19
4.4 Invloed spectrum op gevonden verdeling	22
5 Conclusies en aanbevelingen.....	24
5.1 Conclusies.....	24
5.2 Aanbevelingen	24
Bibliografie	25
Bijlagen	26
Bijlage 1: Matlab code voor het creëren van stochastische velden.....	26
Bijlage 2: Matlab code voor verdeling extreme waarden.....	27
Bijlage 3: Kansverdelingen met een goede fit volgens de Kolmogorov-Smirnov test	28

Samenvatting

Een schaalconstructie wordt gezien als een constructie wat bestaat uit 1 element. Dit element bestaat uit een plaat met minimaal 1 kromme, een zogenoemd schaalement. Door de geometrie van deze soort constructies zijn ze zeer efficiënt in het afdragen van belastingen. Echter zijn schaalconstructies erg gevoelig voor imperfecties waardoor de maximale kniklast drastisch afneemt. De knikweerstand van een schaalconstructie wordt berekend door middel van een eindige elementen methode. Het is hierin noodzakelijk om de imperfecties van de constructie bij zijn initiële vorm te voegen. De enige manier om de realistische imperfecties herhaaldelijk te kunnen simuleren is door ze stochastisch te modelleren. Dit imperfectie veld wordt een zogeheten stochastisch veld genoemd. Dit houdt in dat het een model is waarmee de imperfectie wordt beschreven die varieert in plaats. Er zijn 2 verschillende methode om in een eindige elementen methode bij de berekening van de maximale kniklast de vormimperfecties mee te nemen. De in dit rapport belangrijke methode voegt een vormimperfectie toe in de knikvorm. Om te bepalen hoe groot de amplitude van deze imperfectie is moet de kansverdeling van de maximale waarde van het stochastische veld bekend zijn.

Bij het creëren van een stochastisch veld wordt een sommatie methode van meerdere sinussen toegepast. Het is voor het creëren van een stochastisch veld nodig om het te beschrijven proces stochastisch afhankelijk te maken van 2 dimensies. Dit is mogelijk door een sommatie van een product van sinussen te gebruiken waarin de faseverschillen als stochasten geprogrammeerd worden. De amplitudes en golflengtes van deze sinussen komen voort uit de gemeten imperfecties van reeds gebouwde schaalconstructies met dezelfde bouwmethode. De groepen waarin deze amplitudes en golflengtes verdeeld zijn wordt het spectrum genoemd. Dit spectrum moet uit minimaal 1000 groepen bestaan om ervoor te zorgen dat het gehele stochastische veld zich volgens een normaalverdeling gedraagt. Op deze manier zijn er ook extreem grote imperfecties mogelijk in het model, welke in de realiteit zelfs oneindig groot kunnen zijn.

Voor het vinden van de verdeling van de extreme waarden van de stochastische velden zijn voorgeprogrammeerde functies binnen Matlab gebruikt naast het programma EasyFit wat speciaal bedoeld is voor het vinden van de kansverdeling voor bepaalde data. Bij het vinden van de beste kansverdelingen zijn de geavanceerde verdelingen en verdelingen met meer dan 3 variabelen niet meegenomen. Een verdeling met veel variabelen zorgt ervoor dat hij erg specifiek aangepast kan worden op data. Dit heeft als voordeel dat deze verdeling erg breed inzetbaar is, maar maakt hem ook gevoeliger voor fouten van de gemeten data wat nadelig is voor de te ontwerpen schaalconstructie.

Er zijn drie conclusies die getrokken zijn uit dit onderzoek. Als eerste is gebleken dat de 3 veel gebruikte maximale waarde verdelingen (Gumbel, Fréchet, Weibull) niet van toepassing zijn op de extreme waarde van deze stochastische velden toepassing. De tweede conclusie die getrokken is gaat over de verdelingen die wel een goede benadering van de data zijn. Er is geen duidelijk onderscheid gebleken uit de volgende 5 verdeling: Fatigue Life (Birnbaum-Saunders), Gamma, Lognormal, Log-Pearson 3 en Pearson 6. Als laatste is er ook bevestigd dat deze verdeling niet bepaald worden door de opbouw van het spectrum dat gebruikt wordt voor het creëren van de stochastische velden.

Het verdient aanbeveling om EasyFit of een vergelijkbaar programma te gebruiken met meer dan 5000 datapunten om te onderzoeken of de conclusies over de gevonden kansverdelingen die in dit rapport gemaakt zijn daadwerkelijk correct zijn. Ook is er onderzoek nodig welke verdeling het beste toepasbaar is in de berekeningen met de eindige elementen methode, zodat de berekening nog efficiënter gedaan kan worden.

1 Inleiding

1.1 Probleemstelling

Schaalconstructies zijn erg gevoelig voor imperfecties en hierbij vooral vormimperfecties. Dit is een reden voor het grote verschil tussen de theoretische maximale belasting en de realistische maximale belasting. Hiervoor is het belangrijk dat in het berekenen van een schaalconstructie vormimperfecties meegenomen worden. De imperfecties ontstaan tijdens de bouw of zelfs tijdens de gebruiksfase dus bij het ontwerpen moet hiervoor een aanname gedaan worden, omdat deze nog niet bekend zijn. Dit is belangrijk voor het vinden van de draagkracht. Het is van belang om te weten wat de grootste en kleinste imperfecties zijn. Dit wordt gedaan door het meten van de vormimperfecties bij soortgelijke constructies die met dezelfde bouwmethode gebouwd zijn. Deze vormimperfecties hebben een kansdichtheid van optreden.

1.2 Doelstelling

Het doel van dit rapport is het vinden van de verdeling van extreme waarden van stochastische velden toegepast op vormimperfecties in schaalconstructies.

1.3 Aanpak/leeswijzer

De opbouw van dit rapport is als volgt. In hoofdstuk 2 wordt eerst de benodigde achtergrond informatie gegeven om een goed beeld te krijgen in welke context dit onderzoek uitgevoerd is. Vervolgens wordt in hoofdstuk 3 de theorie achter het geschreven Matlab programma beschreven en wordt de werkwijze van dit programma uitgelegd. In hoofdstuk 4 worden de resultaten van de simulaties in het geschreven programma beschreven. Hoofdstuk 5 bevat de conclusies die uit deze resultaten getrokken kunnen worden met betrekken tot de doelstelling van dit onderzoek.

2 Achterliggende theorie

2.1 Schaalconstructies

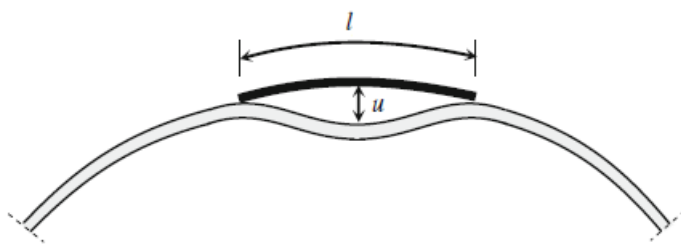
Een schaalconstructie wordt gezien als een constructie wat bestaat uit 1 element. Dit element heeft net als een plaat 1 dimensie die aanzienlijk kleiner is in vergelijking met de andere. Wat een schaal onderscheidt van een plaat is een kromming. Dit kan 1 kromming zijn zoals bij een halve cilinder of meerdere krommingen zoals bij een hypar (hyperbolische paraboloid) dak (Blaauwendraad & Hoefakker, 2014). Door de geometrie van deze soort constructies zijn ze efficiënt in het afdragen van belastingen, het is mogelijk om met kleine diktes toch een grote draagkracht te creëren.

Echter een schaalconstructie heeft ook een groot nadeel, het geeft geen waarschuwing voordat hij bezwijkt (Hoogenboom, 2016). De meeste schaalconstructies gedragen zich op dezelfde manier bij een kniksituatie. Wanneer de kritieke stress in de schaal bereikt is er een plotselinge terugval waardoor de maximale draagkracht van de constructie drastisch vermindert. Imperfecties in een schaalconstructie zorgen ervoor dat de theoretisch bepaalde kritieke spanning niet maatgevend is. Door deze imperfecties is een veel lagere spanning genoeg om de draagkracht te verlagen, wat waarschijnlijk ervoor zorgt dat de constructie bezwijkt.

Het is dus noodzakelijk dat deze constructies goed ontwerpen en gebouwd worden. Het is nagenoeg onmogelijk om een constructie te bouwen zonder enige imperfecties te creëren. Er zijn verschillende soorten imperfecties zoals restspanningen, temperatuur spanningen, niet volledig homogeen materiaal, krimp, excentriciteit van belasting en vormimperfecties. In dit onderzoek is alleen gekeken naar vormimperfecties.

2.2 Vormimperfecties

Met een vormimperfectie wordt een verschil in positie bedoeld tussen de perfecte (ontworpen) schaal en de gerealiseerde schaal, zie figuur 1.



Figuur 1 (Blaauwendraad & Hoefakker, 2014, p. 282): Vormimperfectie met l =lengte, u =amplitude

Vormimperfecties hebben in tegenstelling tot andere imperfecties het voordeel dat ze relatief eenvoudig te simuleren zijn. Door middel van moderne computers zijn deze vormimperfecties eenvoudig te realiseren. Als er genoeg gegevens zijn over imperfecties in reeds gebouwde schaalconstructies die door middel van dezelfde bouwmethode gebouwd zijn is het mogelijk om realistische vormimperfecties te creëren. Hiermee is het mogelijk om een maximale kniklast te berekenen waarin de vormimperfecties meegenomen zijn. Door middel van een zogeheten 'knockdown' factor worden de andere imperfecties die mogelijk aanwezig zijn verwerkt in de maximale kniklast. Dit zorgt uiteindelijk voor een redelijk nauwkeurige benadering van de werkelijk kniklast.

De knikweerstand van een schaalconstructie wordt berekend door middel van een eindige elementen methode. Het is hierin noodzakelijk om de imperfecties van de constructie bij zijn initiële vorm te voegen. Deze imperfecties zorgen er namelijk voor dat de schaalconstructie zich anders gaat gedragen als er een belasting optreedt. Uiteindelijk zorgen deze imperfecties ervoor dat er een kniklast optreedt die lager is dan de kniklast die berekend wordt voor de perfecte (ontworpen) schaalconstructie.

De ideale omstandigheid voor de ontwerpende ingenieur is het weten van de optredende imperfecties waardoor het mogelijk is om de exacte kniklast te kunnen bepalen. Dit is natuurlijk niet mogelijk, omdat de daadwerkelijke constructie nog in de ontwerp fase zit (Schmidt, 2000).

2.3 Modeleren van vormimperfecties

De enige manier om de realistische imperfecties herhaaldelijk te kunnen simuleren is door ze stochastisch te modelleren. Deze manier is gebaseerd op het patroon van de initiële imperfecties. Deze imperfecties zijn ontstaan tijdens het maken van de constructie. Hierdoor hebben deze vormimperfecties een sterke overeenkomst met de constructiemethode. Een bepaalde constructiemethode van de schaalconstructie zorgt voor fouten die karakteristiek zijn voor deze methode, maar er zijn veel meer factoren die meewegen in het ontstaan van vormimperfecties zoals een niet perfect homogeen verdeling in het materiaal of slechte weersomstandigheden tijdens de bouw.

Deze imperfecties worden gezien als random variabelen die dus ook op die manier moeten worden gesimuleerd. Doordat er een kans is op een bepaalde soort vormimperfectie is het berekening van de kniksterkte van een schaalconstructie een probabilistisch probleem geworden (Poggi & Chryssanthopoulos, 1995).

Het verkrijgen van de informatie van eerder gebouwde schaalconstructies is een belangrijk principe in deze methode. Dit wordt gedaan door het meten van deze schaalconstructies. Dit was vroeger altijd een moeilijk proces, omdat er nog niet een goede techniek was waarmee dit nauwkeurig en snel gedaan kon worden. Tegenwoordig is deze techniek er wel, namelijk in de vorm van laserscanners. Deze moderne techniek zorgt ervoor dat met een zeer kleine meetfout bijna alle vormimperfecties gevonden kunnen worden zonder dat de persoon die het apparaat bediend veel fouten kan begaan.

In samenwerking met moderne computerprogramma's is het mogelijk om met deze scanners een goed beeld te krijgen van de vormimperfecties in reeds gebouwde constructies. Het is na het verkrijgen van deze informatie mogelijk om een computer een groot aantal verschillende realistische vormimperfectie velden te laten maken.

Doordat er een kans is op een bepaalde vormimperfecties kunnen de gemeten imperfecties van andere schaalconstructies in combinatie met een random generator in een computerprogramma gebruikt worden om een "nieuw" imperfectie veld te genereren. Dit imperfectie veld wordt een zogeheten stochastisch veld genoemd. Dit houdt in dat het een model is waarmee de imperfectie wordt beschreven die varieert in plaats. Met deze velden is het mogelijk om het complexe probleem van nog niet bekende imperfectie velden in schaalconstructies toch te modeleren.

Er zijn 2 verschillende methoden om in een eindige elementen methode bij de berekening van de maximale kniklast de vormimperfecties mee te nemen, namelijk:

- Toevoegen van een imperfectievorm aan de initiële vorm van de schaalconstructie: stochastisch veld. Dit betekent dat er een stochastisch veld aan de vorm van de schaalconstructie wordt toegevoegd waarna de berekening voor de kniklast wordt gedaan met behulp van de eindige elementen methode. Dit betekent dat er een zeer groot aantal, zeg 10000, niet-lineaire berekeningen gedaan moeten worden. Dit is een erg tijd rovend proces en is daarom niet handig om te gebruiken voor het berekenen van schaalconstructies
- Imperfectie toevoegen in de vorm van de knikvorm. Dit is de maatgevende vorm van een imperfectie als gekeken wordt naar de invloed op de kniklast. Hierbij is het echter wel noodzakelijk om iets te kunnen zeggen over de amplitude van deze imperfectievorm. Hier komt het onderwerp van dit project in het beeld. Deze berekeningen worden gedaan met een lineaire knik analyse waarin de amplitude van de imperfecties een belangrijke rol speelt. Als men de verdeling van de maximale imperfecties weet kan er met een goede zekerheid een kniklast gegeven worden met een daarbij behorende kans waarin dit correct is.

Het grote verschil tussen de twee methodes is het moment waarin de berekening herhaalt wordt. Bij de eerste methode wordt een stochastisch veld gecreëerd en toegevoegd aan de initiële vorm van de schaalconstructie waarna voor dit geval de kniklast wordt berekend. Dit proces wordt daarna herhaald tot er een verdeling ontstaat van de kniklast, waarmee de benodigde maatregelen getrokken kunnen worden.

De Tweede methode maakt geen verdeling van de gevonden kniklast, maar van de extreme waarden van de gecreëerde stochastische velden. Elke herhaling wordt er een stochastisch veld gecreëerd waarna de extreme waarden worden opgeslagen. Hiermee kan een verdeling gevonden worden van deze extreme waarden. De herhaling van het maken van de stochastische velden is minder tijdrovend dan de herhaling van de volledige niet lineaire berekening van de eindige elementen methode.

3 Creëren van stochastische velden

Om te kunnen kijken naar de verdeling van de extreme waarden van een stochastisch veld is het nodig om een groot aantal van deze velden als data te bezitten. In het kader van dit project is er een Matlab code geschreven waarin deze data gemaakt wordt. Dit programma bestaat uit 2 losstaande delen:

- Het eerste programma maakt een stochastisch veld en slaat de maximale en de minimale waarden op.
- Het tweede programma gebruikt de opgeslagen extreme waarden om een kansverdeling te maken.

(De geschreven Codes kunnen gevonden worden in de bijlage)

Het programma waar het stochastische veld wordt gecreëerd is het belangrijkste deel. Het veld moet realistische waarde hebben, anders levert de resterende code informatie die niet klopt met de realiteit en hierdoor geen waarde heeft.

Zoals al eerder genoemd is het herhaaldelijk simuleren van realistische imperfecties alleen mogelijk door dit stochastisch te modelleren, zoals de naam van het stochastische veld al impliceert. Dit heeft betrekking op een random voorkomende gebeurtenis in het model, dit wordt in de Matlab code meegenomen door gebruik te maken van een random generator.

De methode die gebruikt is bij het maken van een stochastisch veld voor de imperfecties is afgeleid van de theorie van random belastingen op constructies (Vrouwenvelder, 2015). In het beschrijven van een random belasting wordt als basis een sinus functie gebruikt met een amplitude, een golflengte en een random faseverschil. Er zijn ook complexere belasting processen te beschrijven met behulp van sinus functies. Wat er gedaan kan worden om deze complexere processen te beschrijven is het sommeren van meerdere sinusfuncties met elk hun eigen amplitude, golflengte en random faseverschil.

Bij het creëren van een stochastisch veld wordt deze sommatie methode van meerdere sinussen toegepast. Echter is het voor het creëren van een stochastisch veld nodig om het te beschrijven proces stochastisch afhankelijk te maken van 2 dimensies. Dit is mogelijk door een sommatie van een product van sinussen te gebruiken:

$$d = \sum a_i * \sin\left(\frac{\pi * x}{l_i} + \zeta_i\right) * \sin\left(\frac{\pi * y}{l_i} + \xi_i\right)$$

Met hierin:

d = waarde van het stochastische veld (mm)

a_i = imperfectie amplitude (mm)

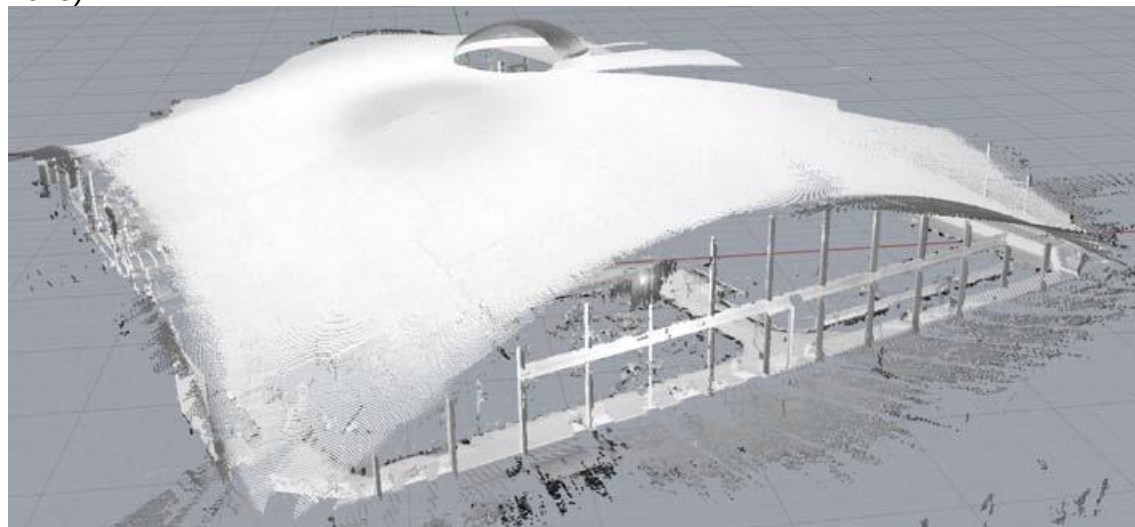
l_i = imperfectie lengte (mm)

ζ_i & ξ_i = random faseverschil, $[0, 2\pi]$

Alleen de faseverschillen zijn stochasten. Deze worden uit een uniforme verdeling van 0 tot 2π getrokken. Deze range is begrenst op 2π aangezien een sinus zich herhaalt na deze periode. De amplitude en de lengte van de imperfecties worden als vaste waarden gebruikt. Dit is gedaan zodat het geschreven programma goed gecontroleerd kan worden met de gebruikte input data (zie §3.1).

De afmetingen van de imperfecties, a_i en l_i , worden gevonden door het meten van al gebouwde schaalconstructies. Dit is gedaan in een afstudeeronderzoek waarbij 4 gewapend

betonnen schaalconstructies ingemeten zijn door middel van een laserscanner (B. Elferink, 2015).



Figuur 2. Puntenwolk van het Heimberg zwembad (B. Elferink, 2015)

In dit afstudeeronderzoek is de gehele constructie ingemeten en met behulp van computerprogramma's is hier een analyse over uitgevoerd. Hieruit zijn de imperfectie grootheden gevonden. Deze zijn geordend in groepen van imperfectielengtes die een gemiddelde imperfectieamplitude hebben (zie onderstaande figuur). Hierdoor gaat er echter wel informatie verloren over de verdeling van de imperfecties. Echter kan hier niet aan ontkomen worden. Om een model te kunnen schrijven die het imperfectieveld beschrijft is het nodig om een aantal sinussen bij elkaar op te tellen. Hiervoor is het nodig de gemeten imperfecties op te delen en dit toe te kennen aan de parameters van de sinusfunctie.

De informatie uit het onderzoek van Elferink over de grootheden van de vormimperfecties van de gemeten schaalconstructies is verdeeld op lengte en amplitude van de imperfectie.

imperfection length [m]	0.0-1.0	1.0-2.0	2.0-3.0	3.0-4.0	4.0-5.0	5.0-6.0
Deitingen petrol station North	6.3	11.2	13.1	15.1	22.9	39.8
Deitingen petrol station South	7.1	9.3	11.3	12.3	23.2	31.4
Heimberg swimming pool	4.9	7.0	6.0	6.1	5.9	4.6
Heimberg tennis hall	3.8	5.8	7.4	3.8	10.8	19.5

Er zijn 6 groepen gemaakt in de imperfectielengte en hierbij zijn de gemiddelde amplitudes gegeven.

Tabel 1: Imperfectieamplitude [mm] van gemeten schaalconstructies met imperfectielengte [m] (B. Elferink, 2015)

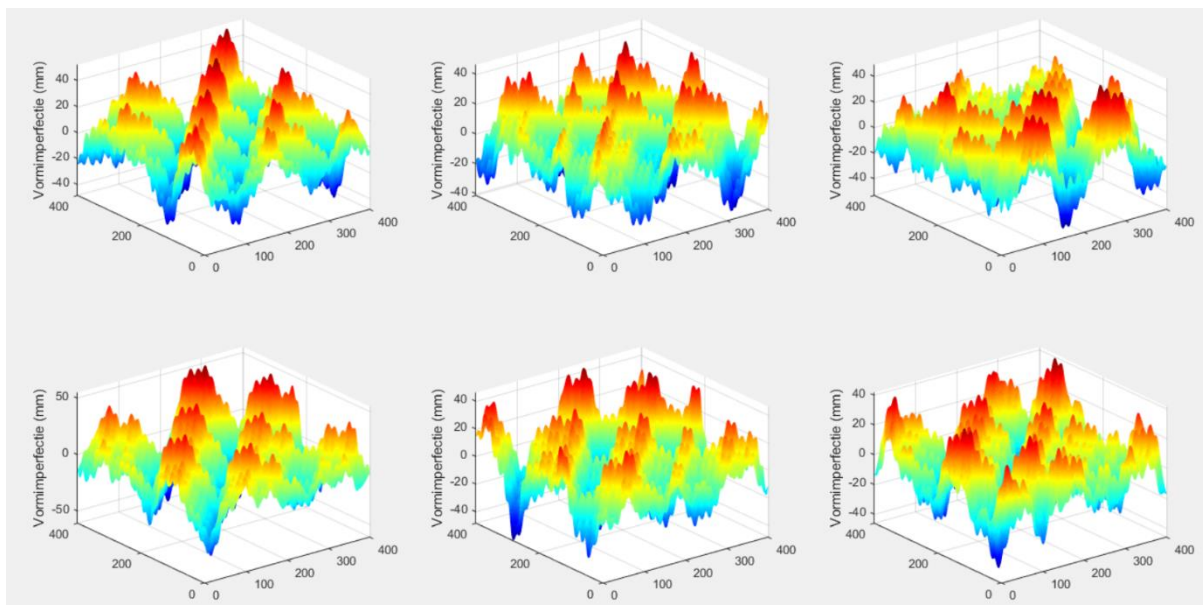
Voor het creëren van de stochastische velden zijn de groepen imperfectielengtes niet in gebieden uitgedrukt, maar is hier het gemiddelde van het gebied gebruikt. Dit is nodig om de data te kunnen gebruiken in het geschreven programma voor het creëren van de stochastische velden. Deze data wordt het spectrum genoemd.

Het tweede programma creëert met de data van de extreme waarden, gegenereerd door het eerste programma, een histogram. Met deze informatie kan de best passende kansverdeling van de extreme waarden gevonden worden. In hoofdstuk 4 wordt er verder ingegaan op de methode waarmee deze kansverdeling gevonden is.

3.1 Controle van het geschreven programma

De geschreven Matlab code moet gecontroleerd worden om te kijken of het maken van de stochastische velden goed gebeurt. Er zijn verschillende eisen waarmee de code gecontroleerd kan worden:

- Elke keer als er een nieuw stochastisch veld wordt gecreëerd, moet deze anders zijn. Dit is een controle om te kijken of de random generator goed meegenomen is in het programma. Als dit niet het geval is wordt de kans op een vormimperfectie niet meegenomen en heeft het onderzoek geen waarde meer.
- De gecreëerde velden moeten eruit zien als een realistische vormimperfectie velden. Hiermee wordt bedoeld dat er geen onrealistische vormen in het veld voorkomen, denk hierbij aan sprongen van de waarde of een volledig nul vlak. De grootte van de imperfectieamplitude moet ook in dezelfde orde zijn als de gemeten schaalconstructies.
- De gemiddelde waarde van het veld moet nul zijn.



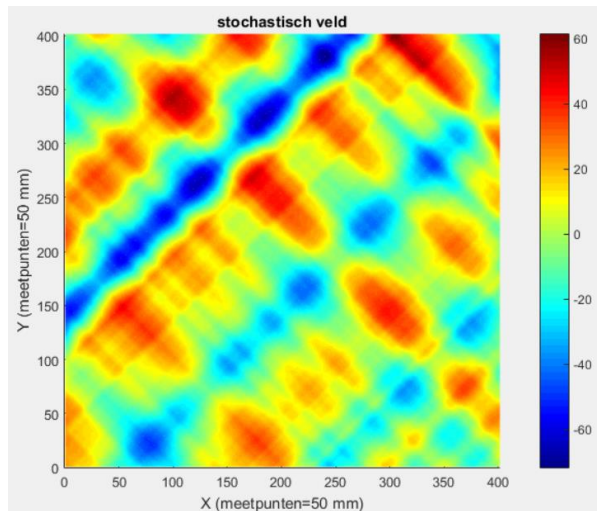
Figuur 3: Aantal met geschreven Matlab gecreëerde stochastische velden

In bovenstaande figuur zijn zes stochastische velden weergegeven die met het gemaakte programma gecreëerd zijn. Het is duidelijk te zien dat elk veld anders is opgebouwd, dus er kan gezegd worden dat de onvoorspelbaarheid van de imperfecties goed meegenomen zijn in het programma door het toevoegen van de random generator voor het faseverschil.

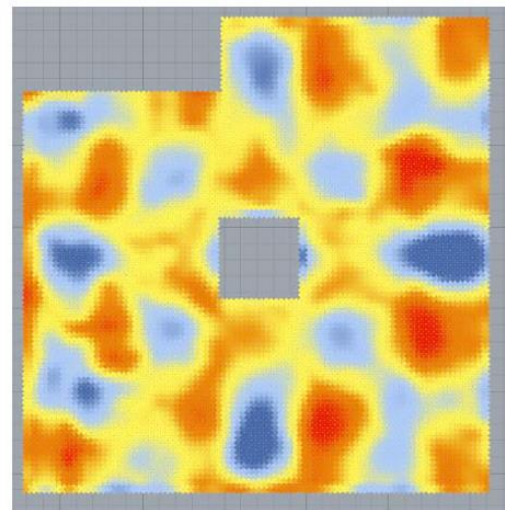
Als er wat beter gekeken wordt naar 1 stochastisch veld is te zien dat er geen onrealistische vormen in het veld aanwezig zijn.

Als er gekeken wordt naar de grootte van de imperfectieamplitude als het veld geconstrueerd wordt met de Deitingen benzinestation komt daar een waarde rond de 75 mm uit. In de meetwaarde van de schaalconstructie is de maximale waarde 80 mm. Dit betekent dat de waarde van de amplitudes ook realistisch is.

Het vergelijken van de gecreëerde stochastische velden met het vormimperfectie veld van een van de gemeten schaalconstructies is ook een indicatie of het programma realistische waarden creëert. In de onderstaande figuren is een gecreëerd stochastisch veld naast de gemeten vormimperfecties van een van de schalen weergegeven. In beide gevallen is te een diagonaal rooster te onderscheiden. Dit geeft aan dat het programma de in werkelijkheid voorkomende vormimperfecties realistisch nabootst.



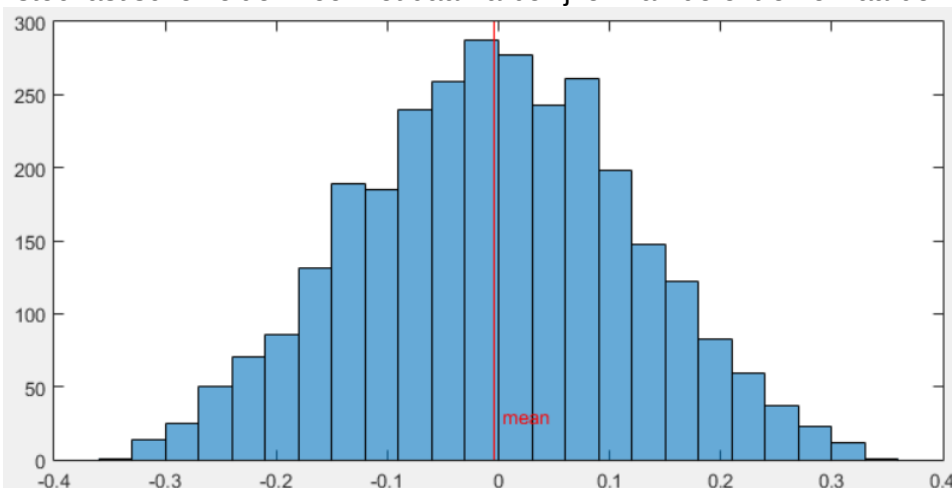
Figuur 5: contourplot van gecreëerd stochastisch veld



Figuur 5: contourplot van gemeten schaalconstructie (B. Elferink, 2015)

Voor de controle of het gemiddelde van het stochastische veld nul is een apart stuk code geschreven. Het is niet mogelijk om elke keer een stochastisch veld te creëren met een gemiddelde van nul door de random faseverschillen die gemaakt worden. Daarom is het nodig om een aantal velden te maken waarna er wel iets zinnigs over de gemiddelde waarde gezegd kan worden, dit is gedaan in de geschreven Matlab code. Deze code genereert een relatief groot aantal stochastische velden en slaat daar de gemiddelde waarde van op. Hiervan is een verdeling gemaakt en die moet een gemiddelde van nul hebben en de standaard afwijking mag hier ook niet ver vandaan liggen.

Zoals te zien in het hieronder gegeven figuur is te zien dat het gemiddelde zeer dicht bij nul ligt en de alle waarde tussen -0.4 en 0.4 mm liggen. Dit zijn relatief kleine waarde als gekeken wordt naar de maximale waarde. Er kan dus met genoeg zekerheid worden gezegd dat het programma een stochastisch veld creëert met een gemiddelde waarde van nul. Het programma voldoet aan alle eisen en is hiermee dus goedgekeurd voor het maken van stochastische velden voor het daarna bekijken van de extreme waarden.



Figuur 6: verdeling van gemiddelde waarde van gecreëerde stochastische velden

4 Verdeling van de extreme waarden

4.1 Gegenerateerde data

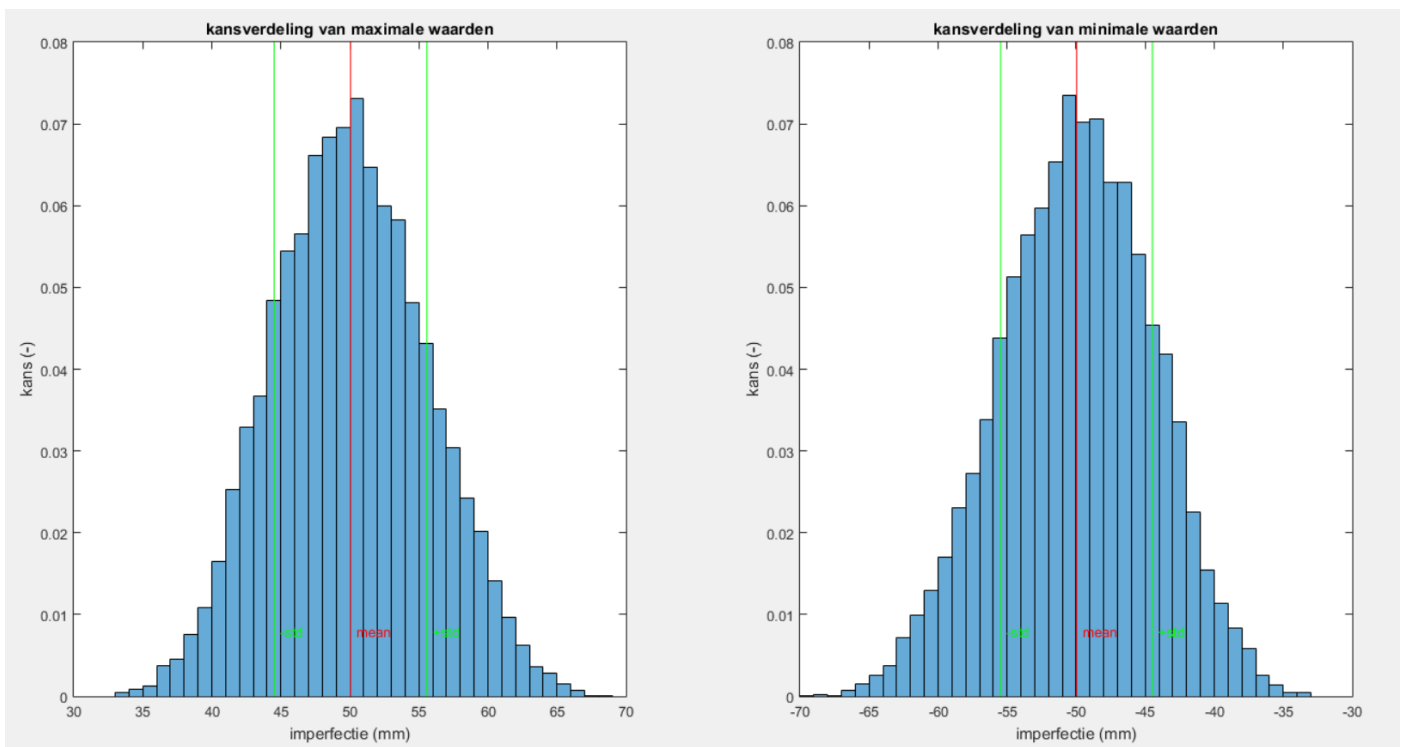
In het voorgaande hoofdstuk is beschreven hoe er met behulp van een zelfgeschreven Matlab code data gecreëerd is over extreme waarden van stochastische velden. Door gebruik te maken van voorgeprogrammeerde functies in Matlab is het mogelijk om karakteristieke van de kansverdeling van deze data te vinden en te gebruiken bij het vinden van de kansverdeling die hierbij hoort.

In eerste instantie zijn de maximale en de minimale waarden apart van elkaar gehouden. Deze 2 verschillende data groepen zijn met elkaar vergeleken. Hieruit is goed te zien dat deze twee groepen in het gemaakte model dezelfde eigenschappen hebben als hun absolute waarde met elkaar worden vergeleken.

Dit is kloppend met de aanname die van te voren gemaakt kan worden als men kijkt naar de manier waarop de stochastische velden gecreëerd worden. De sinussen die in het creëren gesommeerd worden hebben allemaal een evenwichtsstand gelijk aan nul. Als er een groot genoeg aantal stochastische velden wordt gecreëerd zorgt dit ervoor dat de positieve en de negatieve extreme dezelfde verdeling moeten hebben.

	<code>mu</code>	<code>std</code>	<code>grootste</code>	<code>kleinste</code>
Maximale waarde	50.069	5.4489	69.439	34.133
Minimale waarde	-50.038	5.4575	-68.697	-33.671

Tabel 2: Eigenschappen van de maximale en minimale datasets

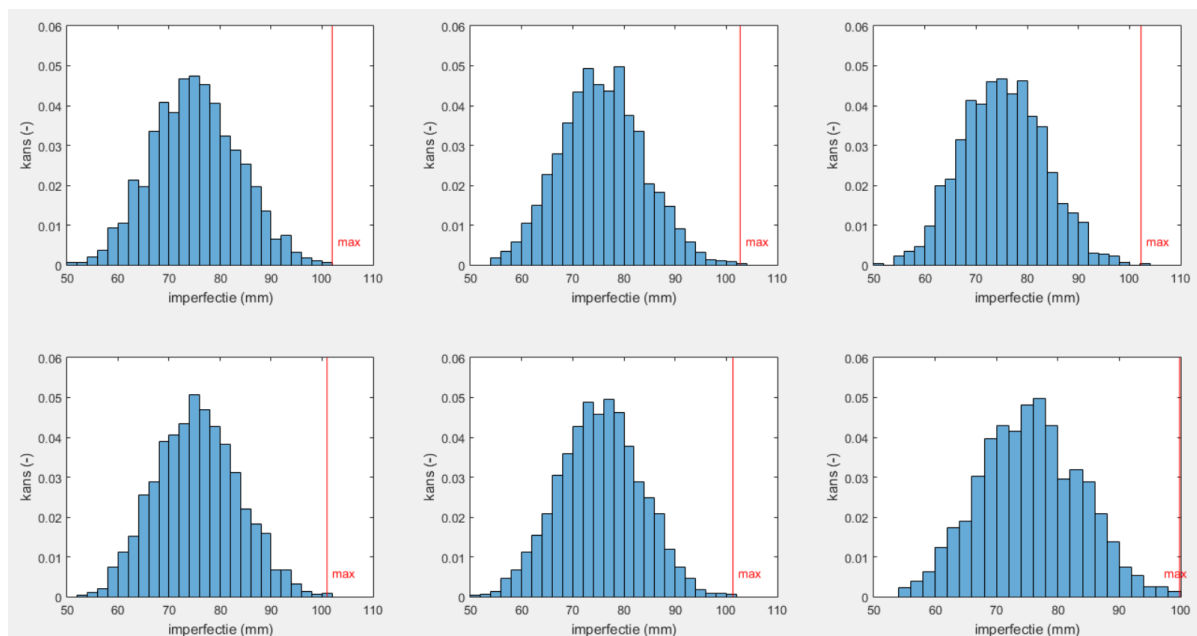


Figuur 7: Histogrammen van maximale en minimale waarden

In het vervolg van dit rapport wordt alleen gekeken naar de verdeling van de maximale waarden. Dit is gedaan zodat de computer berekeningen sneller uitgevoerd kunnen worden, omdat er minder waarden gebruikt hoeven te worden. Dit heeft geen gevolgen voor het uiteindelijke resultaat van dit onderzoek, omdat de positieve en negatieve waarden overeenkomen als er een groot genoeg aantal velden zijn gecreëerd.

Voor het vinden van de best fittende verdeling voor de extreme waarden zijn er 10000 stochastische velden gecreëerd op de in hoofdstuk 3 beschrijven wijze. Dit is een aantal keer gedaan en van deze data zijn histogrammen gemaakt. Dit is gedaan om te kunnen controleren of 10000 stochastische velden genoeg is om de werkelijke vorm van de verdeling te bekijken.

Wat hierbij op viel was de begrenzing van de maximale waarde van de imperfectie. Wanneer de velden werden gemaakt met gebruik van de data van de eerste gemeten schaalconstructie, Deitingen benzinestation noord, was er een grootste waarde in de verdeling die constant in de buurt van 100 mm lag. Dit is weergegeven in onderstaande figuur en tabel.



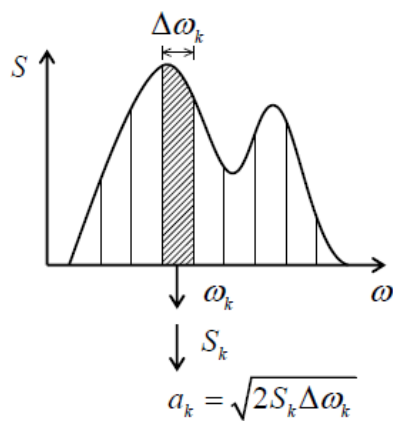
Figuur 8: Verschillende histogrammen gecreëerd met de data van het Deitingen benzinestation noord.

hoogste	
1	101.97
2	102.72
3	102.25
4	100.96
5	101.3

Tabel 3: Hoogste waarde van de 6 histogrammen in figuur 6

Het probleem is een gevolg van de methode waarop de stochastische velden gecreëerd worden. Het spectrum wordt verdeeld in 6 groepen die gebaseerd zijn op de lengte van de imperfecties (zie tabel 1). In de sommatie van alle sinussen is er een maximale imperfectie die ontstaat door alle 6 amplitudes bij elkaar op te tellen. Dit gebeurt als de losse sinussen hun maximale waarden op dezelfde locatie hebben. In de werkelijkheid is de maximale imperfectie die ontstaat niet hieraan gebonden. Ze kunnen in realiteit oneindig groot zijn, echter zal de constructie hierdoor zodanig in draagkracht afnemen dat ze door haar eigen gewicht al bezwijkt.

Om ervoor te zorgen dat de data die gecreëerd wordt door het programma zo realistisch mogelijk te maken is het theoretisch gezien nodig dat de imperfectie oneindig groot moeten kunnen worden. Dit is mogelijk door de sommatie van sinussen te vergroten. Door het aantal sinussen te vergroten wordt het maximum vergroot. Er moet wel een correctie gemaakt worden op de amplitudes zodat er niet alleen extreem grote waarden voor de imperfecties optreden.



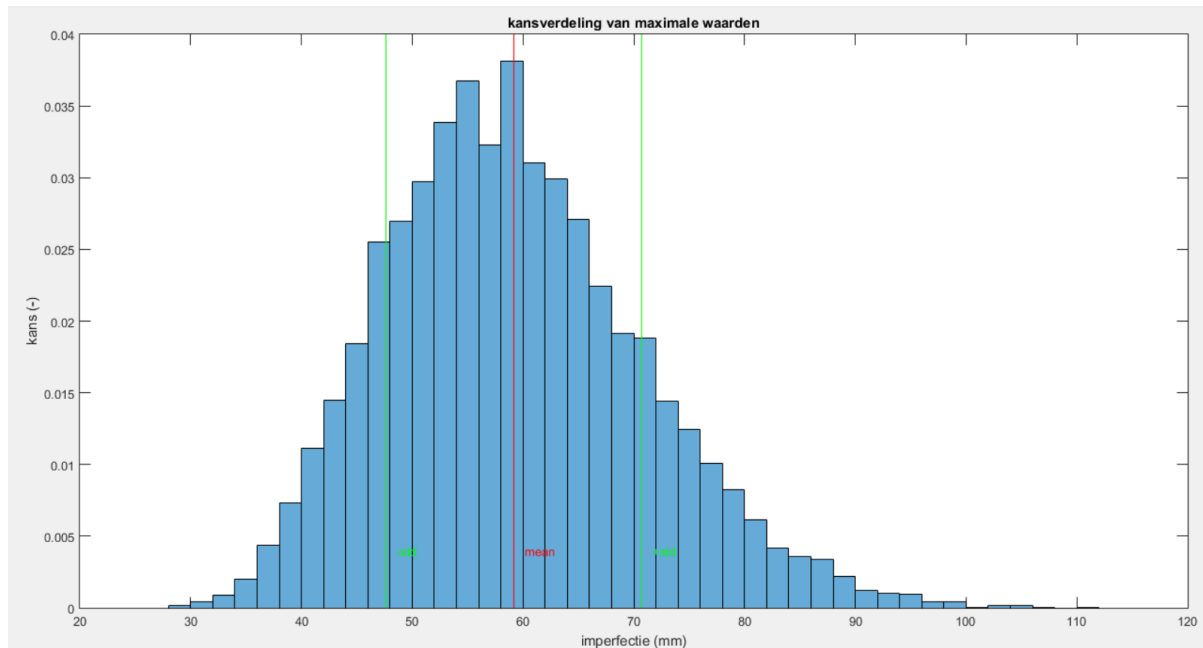
Als de segmenten 2 keer zo klein gemaakt worden is te zien dat de amplitude a_k gedeeld moet worden door $\sqrt{2}$. Dit betekent dat wanneer er N keer zoveel data gebruikt worden dat de amplitudes van de sinussen gedeeld moet worden door $\sqrt{N}$.

figuur 9: random signaal opgebouwd uit segmenten (Vrouwenvelder, 2015)

Een nieuwe complicatie is nu ontstaan doordat er theoretisch gezien een oneindige hoeveelheid sinussen bij elkaar opgeteld moet worden om de mogelijkheid te creëren waarbij een oneindig grote imperfectie ontstaat.

Als deze methode voor het creëren van een stochastisch veld wordt toegepast met een sommatie van minimaal 1000 sinussen gedraagt het gehele veld zich volgens een normaal verdeling (Vrouwenvelder, 2015).

Om ervoor te zorgen dat de sommatie van 6 sinussen uitgebreid kan worden naar een sommatie van minimaal 1000 moet er een aanname gedaan worden voor de waarden binnen de 6 groepen. Door aan te nemen dat de amplitudes binnen de 6 originele groepen lineair verloopt, is het mogelijk om een groter aantal sinussen te sommeren. Door in deze 6 groepen 200 amplitudes onder te verdelen ontstaat er een sommatie van 1200 sinussen en gedraagt het veld zich volgens een normaal verdeling.



Figuur 10: Histogram van data gecreëerd door verbeterde code

De hoogste waarde die de maxima aan kon nemen in de eerste code was 108,4 mm. In de verbeterde code is dit gegroeid naar 1251,6 mm. Hierbij is de kans dat er op 1 plek alle sinussen hun hoogste punt hebben een stuk kleiner dus de kans dat de hoogste waarde wordt behaald is kleiner. Dit is conform de werkelijkheid.

Doordat er nu een veel hogere grens ligt op de maximale imperfectie is er meer data beschikbaar over de staart van de kansverdeling. In de civiele sector is het nodig om met gebeurtenissen te werken die een kleine kans hebben om voor te komen. Dit wordt gedaan om ervoor te zorgen dat er minder ongelukken gebeuren, zoals bijvoorbeeld het instorten van een gebouw door een storm. Dit geldt ook voor de kans dat er een vormimperfectie aanwezig is in een schaalconstructie.

Voor het vinden van een kansverdeling is het dus erg belangrijk dat de staart van de verdeling goed aansluit op de data.

4.2 Het vinden van de best fittende verdeling: Matlab

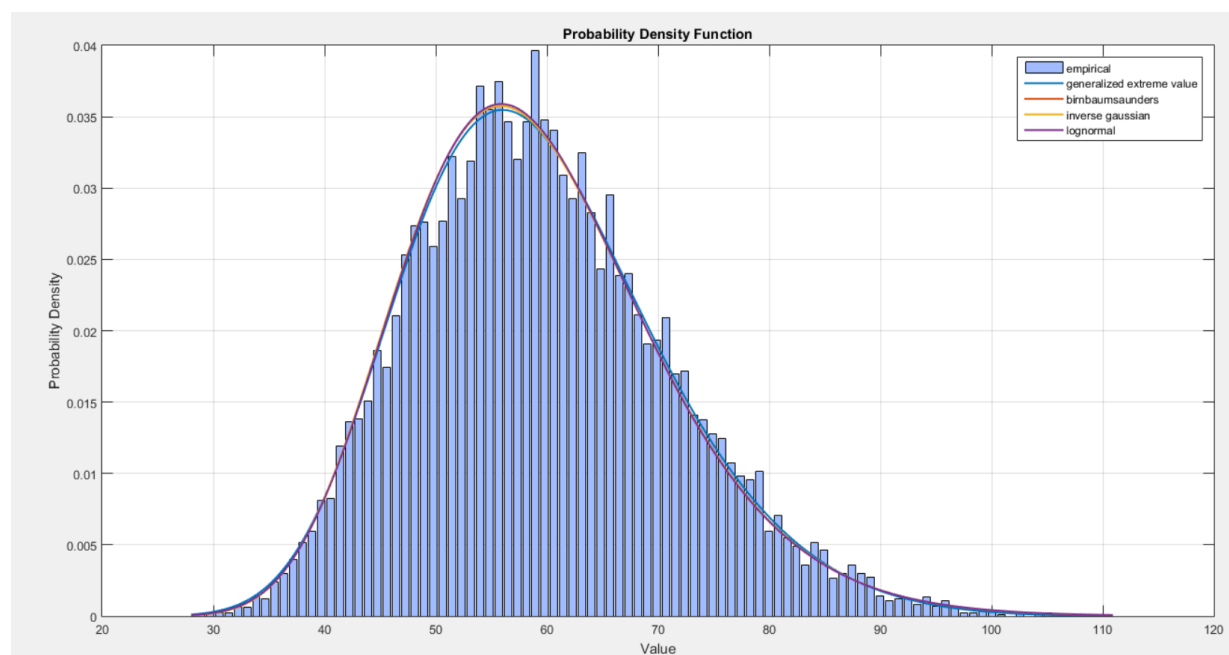
Description
Beta distribution
Binomial distribution
Birnbaum-Saunders distribution
Burr distribution
Exponential distribution
Extreme Value distribution
Gamma distribution
Generalized Extreme Value distribution
Generalized Pareto distribution
Half-normal distribution
Inverse Gaussian distribution
Kernel distribution
Logistic distribution
Loglogistic distribution
Lognormal distribution
Nakagami distribution
Negative Binomial distribution
Normal distribution
Poisson distribution
Rayleigh distribution
Rician distribution
Stable distribution
t Location-Scale distribution
Weibull distribution

Matlab heeft veel voorgeprogrammeerde functies die gebruikt kunnen worden door ze simpelweg aan te roepen. Hieronder valt ook de functie waarmee een kansverdeling gefit kan worden over de data, de zogenoemde *fitdist* functie. Deze functie zoekt de parameters waarbij de functie van de kansverdeling het beste aansluit op de data zelf.

Om de verschillende kansverdelingen te kunnen onderscheiden van elkaar is de functie *allfitdist* gebruikt (Sheppard, 2012). Deze functie maakt gebruik van alle voorgeprogrammeerde kansverdelingen in Matlab. De functie fit alle kansverdelingen en rangschikt ze van beste naar slechtste fit. Als laatste plot deze functie de data samen met de 4 best fittende verdelingen. In Matlab zijn ook een aantal testen opgenomen om te controleren of een verdeling een goeie fit heeft met de data, deze zijn gebruikt om de rangschikking te maken.

In figuur 10 is de plot van de top 4 best fittende kansverdelingen weergegeven over de maximale waarden van 10000 stochastische velden. Op het oog is er zeer weinig verschil te ontdekken tussen de verdelingen. Het is dus nodig om een test uit te voeren of de kansverdeling goed genoeg. Hiervoor zijn verschillende testen ter beschikking, in dit rapport is gebruik gemaakt van de Kolmogorov-Smirnov test.

Figuur 11:
voorgeprogrammeerde
kansverdelingen van Matlab
(The Mathworks Inc, 2016)



Figuur 12: plot van de best fittende voorgeprogrammeerde verdeling

De volgens *allfitdist* beste 8 verdelingen zijn getest met behulp van de Kolmogorov-Smirnov test. Deze test is ook voorgeprogrammeerd in Matlab en stelt de nul hypothese dat de data uit de geteste verdeling komt. Dit wordt getest tegen de alternatieve hypothese dat dit niet het geval is met een zekerheid van 5%. Als uitkomst geeft de test een waarde van 0 of 1 aan h . Deze waarde van h betekenen dat de nul hypothese niet of wel verworpen wordt. Uit de test blijkt dat alleen de eerste 5 verdeling voldoen.

	nummer	h
	-----	---
GeneralizedExtremeValue	1	0
BirnbaumSaunders	2	0
InverseGaussian	3	0
Lognormal	4	0
Gamma	5	0
Nakagami	6	1
Loglogistic	7	1
Rician	8	1

Tabel 4: KS-test voor de beste 8 verdelingen volgens Matlab

De Generalized Extreme Value kansverdeling is een geavanceerde kansverdeling die opgebouwd is uit de 3 kansverdeling die gebruikt worden bij het beschrijven van maximale waarden. Dit zijn de Gumbel, Fréchet en Weibull verdelingen. Het samenvoegen van deze 3 verdelingen zorgt ervoor dat de verdeling 3 variabelen nodig heeft en erg gevoelig is voor de data waarover hij gefit wordt. Dit laatste betekend dat de verdeling goed aan kan sluiten op de data, maar dit zorgt voor een groot nadeel. Als de gemeten data van reeds gebouwde schaalconstructies fouten bevatten fit deze kansverdeling hier heel nauwkeurig overheen. Hierdoor hebben deze fouten erg veel invloed op de uiteindelijke gegevens die gebruikt worden voor het berekenen van de nieuwe schaalconstructie die wordt ontworpen. Om dit te voorkomen is het niet gewenst deze geavanceerde kansverdelingen te gebruiken voor het berekenen van schaalconstructies.

Matlab heeft de 3 maximale waarde verdelingen niet voorgeprogrammeerd. Hierdoor is het nodig om een ander programma te vinden die deze verdelingen wel kan testen. In de volgende paragraaf is een programma gevonden die dit kan.

4.3 Het vinden van de best fittende verdeling: EasyFit

EasyFit (MathWave Technologies, 2015) is een programma dat speciaal gemaakt is voor het analyseren van probabilistische data. Het heeft een grotere database dan Matlab, namelijk 60 verschillende verdelingen, die gefit kunnen worden op de data die je het programma toe dient. In deze database zitten ook de 3 maximale waarde verdelingen (de Gumbel, Fréchet en Weibull verdelingen).

Het programma beschikt ook over verschillende ingebouwde testen voor het beschouwen van de gefitte verdelingen. Om gevonden verdeling goed te kunnen vergelijken met de uitkomsten van Matlab is ook hier gebruik gemaakt van de Kolmogorov-Smirnov test.

EasyFit heeft een grote database van verdelingen, echter zijn hier veel geavanceerde verdelingen in opgenomen. Hiermee worden verdelingen met veel variabelen bedoelt of verdelingen die opgebouwd zijn uit een combinatie van andere kansverdelingen zoals bij de *Generalized Extreme Value* verdeling het geval is.

Een verdeling met veel variabelen zorgt ervoor dat hij erg specifiek aangepast kan worden op data. Dit heeft als voordeel dat deze verdeling erg breed inzetbaar is, omdat hij bij veel verschillende data aangepast kan worden en een mooie fit levert. Echter zijn deze verdelingen net als zeer geavanceerde kansverdelingen erg gevoelig voor de gebruikte data. Hierdoor zullen deze verdeling minder toegevoegde waarde hebben in de context van dit onderzoek en worden hierom niet meegenomen in de keuze van de verdeling voor de extreme waarden.

Het programma EasyFit heeft echter wel een nadeel ten opzichte van Matlab. Het aantal datapunten dat gebruikt kan worden door het programma is gelimiteerd op 5000. Het is door dit kleinere aantal data punten mogelijk dat er een deel van de echte verdeling niet gevonden wordt en het programma een andere verdeling beter kan fitten. Dit betekent dat er met mindere zekerheid gezegd kan worden of de gevonden beste verdeling ook in werkelijkheid de beste is. Om dit op te lossen is er 4 keer een dataset gemaakt van maximale waarden van 5000 stochastische velden en zijn deze datasets door EasyFit geanalyseerd.

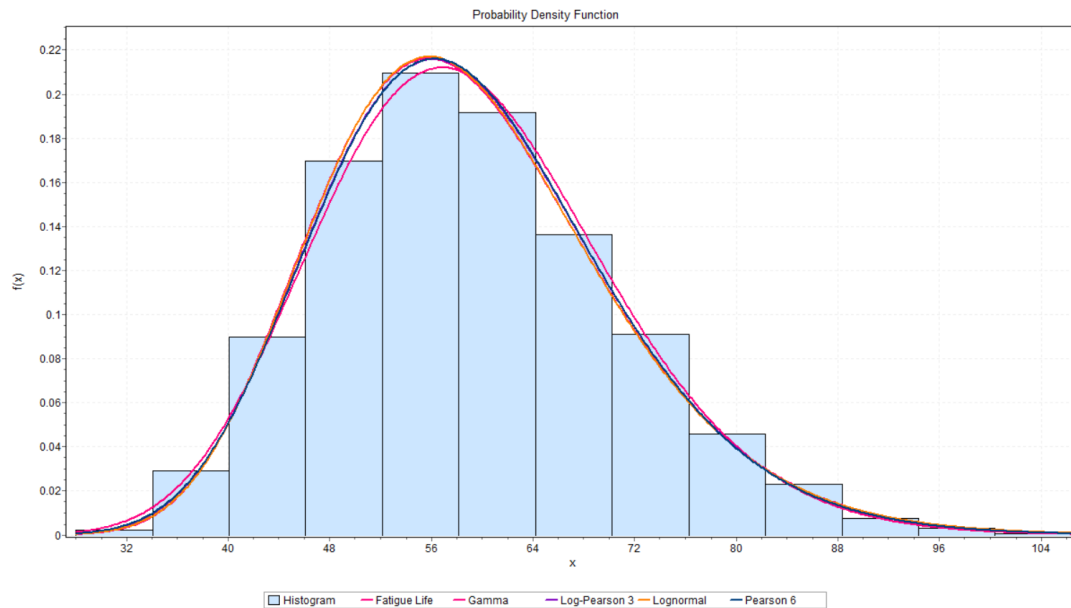
Zoals in Matlab ook naar voren kwam uit het fitten van verschillende kansverdelingen zijn er meerdere goed fittende kansverdelingen. De kansverdelingen die volgens de Kolmogorov-Smirnov test een goede fit hebben zijn hieronder weergegeven. Hierbij zijn de geavanceerde verdeling en verdelingen met meer dan 3 variabelen niet meegenomen. Fatigue Life verdeling wordt ook wel de Birnbaum-Saunders genoemd. (In bijlage 3 zijn alle verdelingen met een goede fit opgenomen)

Uit de analyse van de 4 datasets zijn 6 kansverdelingen gekomen die op minimaal 1 van de datasets een goede fit hebben. Bij 5 van de 6 kansverdelingen komt deze verdeling meerdere keren naar voren als een goede fit. Alleen de Log-Gamma kansverdeling komt maar 1 keer als goede kansverdeling uit het programma. Bij de andere 3 datasets wordt de Log-Gamma aangegeven als een slecht fittende kansverdeling. Hierdoor kan worden gezegd dat deze kansverdeling toevallig goed fit bij deze dataset. Dit is ontstaan door de relatief kleine datasets van 5000 punten. De Log-Gamma kansverdeling wordt hierdoor afgeschreven als een goede kansverdeling voor de maximale waarden van de stochastische velden.

De verdelingen moet ook visueel een goede fit hebben. Het is mogelijk dat er op een aantal plekken de verdeling niet goed fitten maar over de hele fit heen wel een mooie waarde geven. Bij deze visuele check is zoals al eerder beschreven de staart van de verdeling erg belangrijk dus hier wordt de meeste waarde aan gehecht.

Kansverdeling	Aantal keer goed fittend
Fatigue Life	2/4
Gamma	4/4
Log-Gamma	1/4
Lognormal	2/4
Log-Pearson 3	4/4
Pearson 6	4/4

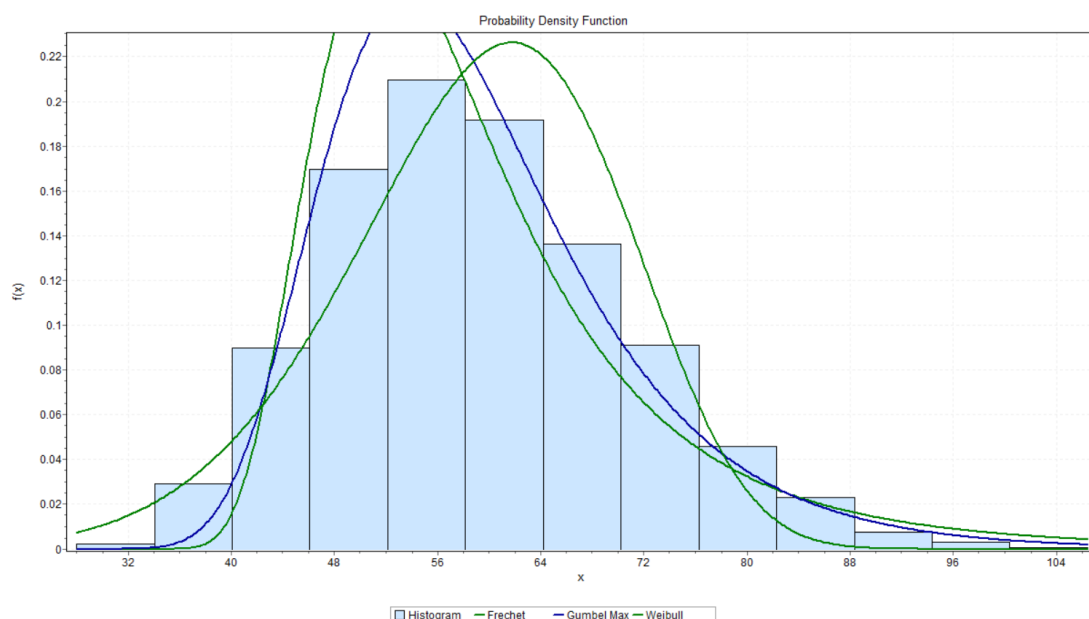
Tabel 5: Goed fittende kansverdeling over de 4 datasets



Figuur 13: Histogram van 1 van de 4 datasets met de 5 best fittende kansverdelingen d.m.v. EasyFit

Bij alle 4 de datasets is te zien dat alle verdelingen erg goed de histogram volgen, zoals verwacht wordt van een goed fittende kansverdeling, zie figuur 11. Een tweede opvallend resultaat heeft te maken met de staarten van de verdeling. De staarten van de verdeling zijn bijna identiek aan elkaar en fitten ook in de staart van de verdeling erg goed. Het is hierdoor niet mogelijk om een van de verdelingen als beste te kiezen.

Een opvallende waarneming die volgt uit de resultaten van het fitten van de kansverdelingen is het feit dat de 3 veel gebruikte maximale waarde verdeling niet als goede fit naar voren komen. In Figuur 12 is te zien dat de Gumbel, Fréchet en de Weibull verdeling geen mooie fit creëren.

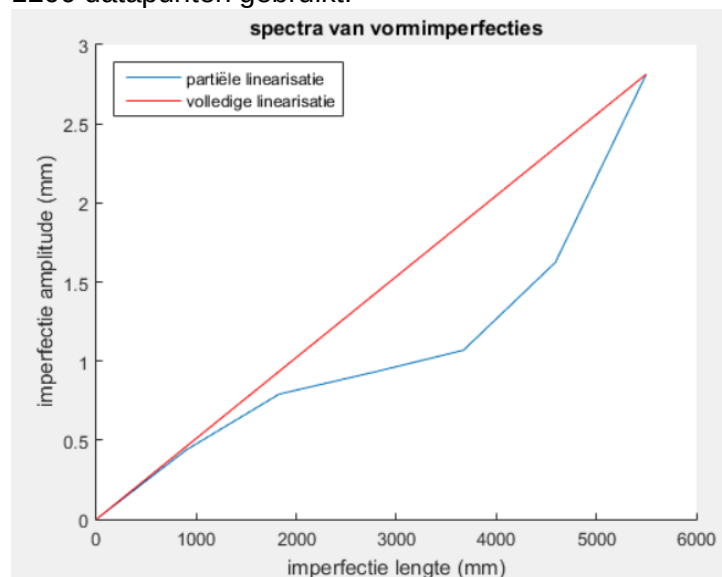


Figuur 14: Fit van Fréchet, Gumbel en Weibull verdelingen

4.4 Invloed spectrum op gevonden verdeling

Door de aanpassing van het spectrum dat gebruikt is voor het creëren van de stochastische velden (zie §4.1) is er mogelijk een verkeerd resultaat ontstaan bij het fitten van de verdelingen. Om te controleren of de opbouw van het spectrum invloed heeft op de uiteindelijk gevonden verdelingen is er nog een verandering aan het spectrum gemaakt waarna weer dezelfde stappen doorlopen zijn voor het vinden van de beste verdelingen.

Voor het vergroten van de datapunten in het spectrum was er in §4.1 een lineair verband tussen de 6 gemeten imperfectie groepen aangenomen. Hiermee wordt er zo veel mogelijk informatie van de daadwerkelijke schaalconstructie met zijn vormimperfecties behouden. Om te controleren of dit verschil maakt met een spectrum dat dit niet doet is een volledige linearisatie van het spectrum gemaakt, zie figuur 13. Bij deze volledige linearisatie zijn ook 1200 datapunten gebruikt.



Figuur 15: Spectra voor creëren van stochastische velden

Als eerste is de data van 10000 stochastische velden in Matlab geanalyseerd.

	nummer	h
	—	—
BirnbaumSaunders	1	0
InverseGaussian	2	0
Lognormal	3	0
Gamma	4	0
GeneralizedExtremeValue	5	0
Nakagami	6	1
Loglogistic	7	1
Rician	8	1

Figuur 16: Best fittende verdeling volgens Matlab (volledig gelineariseerd spectrum)

Matlab geeft hier weer dezelfde verdeling aan als goed fittende verdelingen. De volgorde van de beste verdeling is net anders, maar dit geeft geen duidelijk conclusie. De data is nu net iets anders en dezelfde verdelingen zijn weer als beste uit de analyse gekomen. Dit geeft aan dat volgens Matlab er geen andere verdeling ontstaat als het spectrum anders opgebouwd is.

Kansverdeling	Aantal keer goed fittend
Erlang	1/4
Fatigue Life	2/4
Gamma	4/4
Lognormal	3/4
Nakagami	1/4
Pearson 6	3/4

Figuur 17: Best fittende kansverdeling volgens EasyFit (volledig gelineariseerd spectrum)

Om dit resultaat te bevestigen is de data ook door het programma EasyFit geanalyseerd. Dit is op exact dezelfde manier uitgevoerd als eerder beschreven. In figuur 15 is te zien dat er 2 verdelingen zijn die 1 keer voorkomen bij de 4 datasets. Het is zelfs het geval dat deze bij dezelfde dataset zijn gevonden. De verdelingen Erlang en Nakagami worden op grond van een niet volledige dataset niet meer meegenomen in de verdere analyse.

De verdelingen die uit de volledige linearisatie komen zijn uiteindelijk:

- Fatigue Life verdeling
- Gamma verdeling
- Lognormal verdeling
- Pearson 6 verdeling

Deze 4 verdeling zijn ook uit de partiële linearisatie gekomen. Dit bevestigt het resultaat uit Matlab.

5 Conclusies en aanbevelingen

5.1 Conclusies

Het doel van dit rapport is het vinden van de verdeling van extreme waarden van stochastische velden toegepast op vormimperfecties in schaalconstructies. Het onderzoek hiervoor heeft de volgende resultaten opgeleverd:

1. De vormimperfecties in schaalconstructies, die stochastisch gesimuleerd worden in een zogeheten stochastisch veld kunnen niet goed beschreven worden met een van de 3 maximale waarde verdelingen. Dit zijn de veel gebruikte Gumbel, Fréchet en Weibull verdelingen.
2. De vormimperfecties in schaalconstructies kunnen door veel verschillende kansverdelingen beschreven worden. Onder deze grote groep verdelingen zijn echter veel verdelingen met een groot aantal variabelen. Hierdoor zijn ze niet goed toepasbaar in het berekenen van schaalconstructies, omdat ze erg gevoelig zijn voor fouten in de benodigde gemeten data van reeds gebouwde schaalconstructies. Als alleen wordt gekeken naar relatief eenvoudige verdelingen met niet meer dan 3 variabelen blijven er 5 verdelingen over die de data goed beschrijven. Deze 5 verdelingen zijn:
 - **Fatigue Life verdeling**, ook wel **Birnbaum-Saunders** verdeling genoemd
 - **Gamma verdeling**
 - **Lognormal verdeling**
 - **Log-Pearson 3 verdeling**
 - **Pearson 6 verdeling**
3. De opbouw van het spectrum van de vormimperfecties die gebruikt wordt bij het creëren van de stochastische velden heeft geen invloed op de uiteindelijk gevonden kansverdelingen voor de extreme waarden.

5.2 Aanbevelingen

Het gebruik van het programma EasyFit heeft zijn voordelen bewezen. Echter verdient het aanbeveling om dit of een vergelijkbaar programma te gebruiken met meer dan 5000 datapunten om te onderzoeken of de conclusies over de gevonden kansverdelingen die in dit rapport gemaakt zijn daadwerkelijk correct zijn.

Het is gebleken dat de opbouw van het gebruikte spectrum geen invloed heeft op de kansverdeling van de maximale waarden. Echter zijn hiermee nog niet alle mogelijke variabelen die invloed kunnen hebben op de kansverdeling bekeken. Hoe hebben de oppervlak van de schaal en de grootste imperfectielengte van het spectrum invloed op de gemiddelde waarde en standaardafwijking van de maximale waarden. Dit kan onderzocht worden door een van deze variabelen te veranderen terwijl de andere constant gehouden wordt. Er kan dan worden geanalyseerd hoe de gemiddelde waarde en standaardafwijking veranderen onder invloed van deze variabelen en of er hierdoor een andere kansverdeling gevonden wordt.

Bibliografie

- B. Elferink, P. E. (2015). *Shape imperfections of reinforced concrete shell roofs*. Delft: Delft University of Technology.
- Blaauwendraad, J., & Hoefakker, J. H. (2014). *Structural Shell Analyse*. Dordrecht: Springer.
- Blaauwendraad, J., & Hoefakker, J. H. (2014). Reducing Effect of Imperfections. In *Structural Shell Analyses* (p. 282). Dordrecht: Springer.
- Hoogenboom, P. (2016). *Notes on Shell Structures*. Delft: Delft university of Technology.
- MathWave Technologies. (2015). EasyFit.
- Poggi, C., & Chryssanthopoulos, M. K. (1995). Stochastic Imperfection Modelling in Shell Buckling. *Thin walled structures* 23, pp. 179-200.
- Schmidt, H. (2000). Stability of steel shell structures. *Journal of constructional steel research*, 159–181.
- Sheppard, M. (2012, april 4). *file exchange Mathworks*. Opgehaald van Mathworks: <http://www.mathworks.com/matlabcentral/fileexchange/34943-fit-all-valid-parametric-probability-distributions-to-data>
- The Mathworks Inc. (2016, mei). Matlab. Opgehaald van <http://nl.mathworks.com/>
- Vrouwenvelder, A. (2015). Random Vibrations. In A. Vrouwenvelder, *Dictaat CIE5145 Delft University of* (p. hoofdstuk 2). Delft.

Bijlagen

Bijlage 1: Matlab code voor het creëren van stochastische velden

```
function [exth, extl] = stoveld( a )
%functie maakt stochastisch veld en slaat de maximale en minimale waarden
%op
%   Input:
%       %a = imperfectie amplitudes bij lengtes van 0.5 tot 5.5 met stappen
%           %van 1.0
%   Output:
%       %exth = maximale waarde van het stochastische veld
%       %extl = minimale waarde van het stochastische veld
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
l=[0.5, 1.5, 2.5, 3.5, 4.5, 5.5]*1000; %%mm
N = length(a);
zeta = 2.*pi.*rand(N);
eps = 2.*pi.*rand(N);

[X,Y] = meshgrid(0:50:20000);

d = meshgrid(zeros(1,length(Y)));
for k=1:N
    z=a(k).*sin((pi*X)/l(k)+zeta(k)).*sin((pi*Y)/l(k)+eps(k));
    d = d+z;
end
exth = max(max(d));
extl = min(min(d));
end
```

Bijlage 2: Matlab code voor verdeling extreme waarden

```
clear all
clc
tic
%gemeten data
%%Deitingen petrol station North%%
a1=[6.3, 11.2, 13.1, 15.1, 22.9, 39.8]; %%mm
%%Deitingen petrol station South%%
a2=[7.1, 9.3, 11.3, 12.3, 23.2, 31.4]; %%mm
%%Heimberg swimming pool%%
a3=[4.9, 7.0, 6.0, 6.1, 5.9, 4.6]; %%mm
%%Heimberg tennis hall%%
a4=[3.8, 5.8, 7.4, 3.8, 10.8, 19.5]; %%mm

a = a1;

T = 3000; %%aantal te maken stochastische velden
%%arrays voor het opslaan van de extreme waarden
exth = zeros(1,T);
extl = zeros(1,T);

for i=1:T
    [exth(i),extl(i)]=stoveld(a);
end

%%De maximale waarde verdeling%%
muh = mean(exth)
sth = std(exth)

figure
subplot(1,2,1)
nbins=40;
histogram(exth,nbins,'Normalization','probability')
hold on
vline(muh,'r-','mean');
title('kansverdeling van maximale waarden')
xlabel('imperfectie (mm)')
ylabel('kans (-)')
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%De minimale waarde verdeling%%
mul = mean(extl)
stl = std(extl)

%figure
subplot(1,2,2)
nbins=40;
histogram(extl,nbins,'Normalization','probability')
hold on
vline(mul,'r-','mean');
title('kansverdeling van minimale waarden')
xlabel('imperfectie (mm)')
ylabel('kans (-)')
toc
```

Bijlage 3: Kansverdelingen met een goede fit volgens de Kolmogorov-Smirnov test

Dataset 1				Dataset 2			
#	Distribution	Kolmogorov Smirnov		#	Distribution	Kolmogorov Smirnov	
		Statistic	Rank			Statistic	Rank
18	Gen. Extreme Value	0.00628	1	21	Gen. Extreme Value	0.00644	1
31	Log-Pearson 3	0.00682	2	23	Gen. Gamma (4P)	0.0071	2
14	Fatigue Life (3P)	0.00682	3	1	Beta	0.00799	3
39	Pearson 5 (3P)	0.00785	4	20	Gamma (3P)	0.0083	4
33	Lognormal	0.01007	5	49	Pearson 6 (4P)	0.00853	5
13	Fatigue Life	0.01013	6	15	Fatigue Life	0.00881	6
17	Gamma	0.01081	7	30	Johnson SB	0.00903	7
				16	Fatigue Life (3P)	0.00908	8
				38	Log-Pearson 3	0.00933	9
				40	Lognormal	0.00943	10
				41	Lognormal (3P)	0.00965	11
				48	Pearson 6	0.00968	12
				47	Pearson 5 (3P)	0.01014	13
				3	Burr (4P)	0.01332	14
				35	Log-Gamma	0.01336	15
				19	Gamma	0.01391	16
				22	Gen. Gamma	0.01438	17
Dataset 3				Dataset 4			
#	Distribution	Kolmogorov Smirnov		#	Distribution	Kolmogorov Smirnov	
		Statistic	Rank			Statistic	Rank
30	Johnson SB	0.0052	1	30	Johnson SB	0.00724	1
1	Beta	0.00562	2	1	Beta	0.00899	2
21	Gen. Extreme Value	0.00728	3	21	Gen. Extreme Value	0.01049	3
23	Gen. Gamma (4P)	0.00742	4	23	Gen. Gamma (4P)	0.01067	4
20	Gamma (3P)	0.00901	5	20	Gamma (3P)	0.01208	5
38	Log-Pearson 3	0.00915	6	38	Log-Pearson 3	0.01282	6
16	Fatigue Life (3P)	0.00982	7	16	Fatigue Life (3P)	0.01297	7
41	Lognormal (3P)	0.0104	8	48	Pearson 6	0.01343	8
48	Pearson 6	0.01046	9	41	Lognormal (3P)	0.01362	9
3	Burr (4P)	0.01047	10	19	Gamma	0.01381	10
19	Gamma	0.01054	11	49	Pearson 6 (4P)	0.01415	11
22	Gen. Gamma	0.01057	12	47	Pearson 5 (3P)	0.01416	12
49	Pearson 6 (4P)	0.0108	13	22	Gen. Gamma	0.01437	13
47	Pearson 5 (3P)	0.01091	14	6	Chi-Squared (2P)	0.01461	14