

Dokumentation MAXFAT

9. März 2022

Inhaltsverzeichnis

1	Einleitung	2
2	Lebensdauerrechnung	4
2.1	Definition der Schnittwinkel	4
3	Kerbsimulation	7
4	Schädigungsparameter	10
4.1	Smith-Watson-Topper P_{RAM}	10
4.2	Fatemi-Socie P_{FS}	12
4.3	Kurzrissmodell P_Z	14
5	Abschätzen von Kennwerten	17
6	Beispielrechnung	19

Kapitel 1 Einleitung

Dieses Dokument dient als Beispiel für die Anwendung des hier zur Verfügung gestellten Codes. Das Programmpaket folgt im wesentlichen dem in Abbildung 1.1 dargestellten Verlauf.

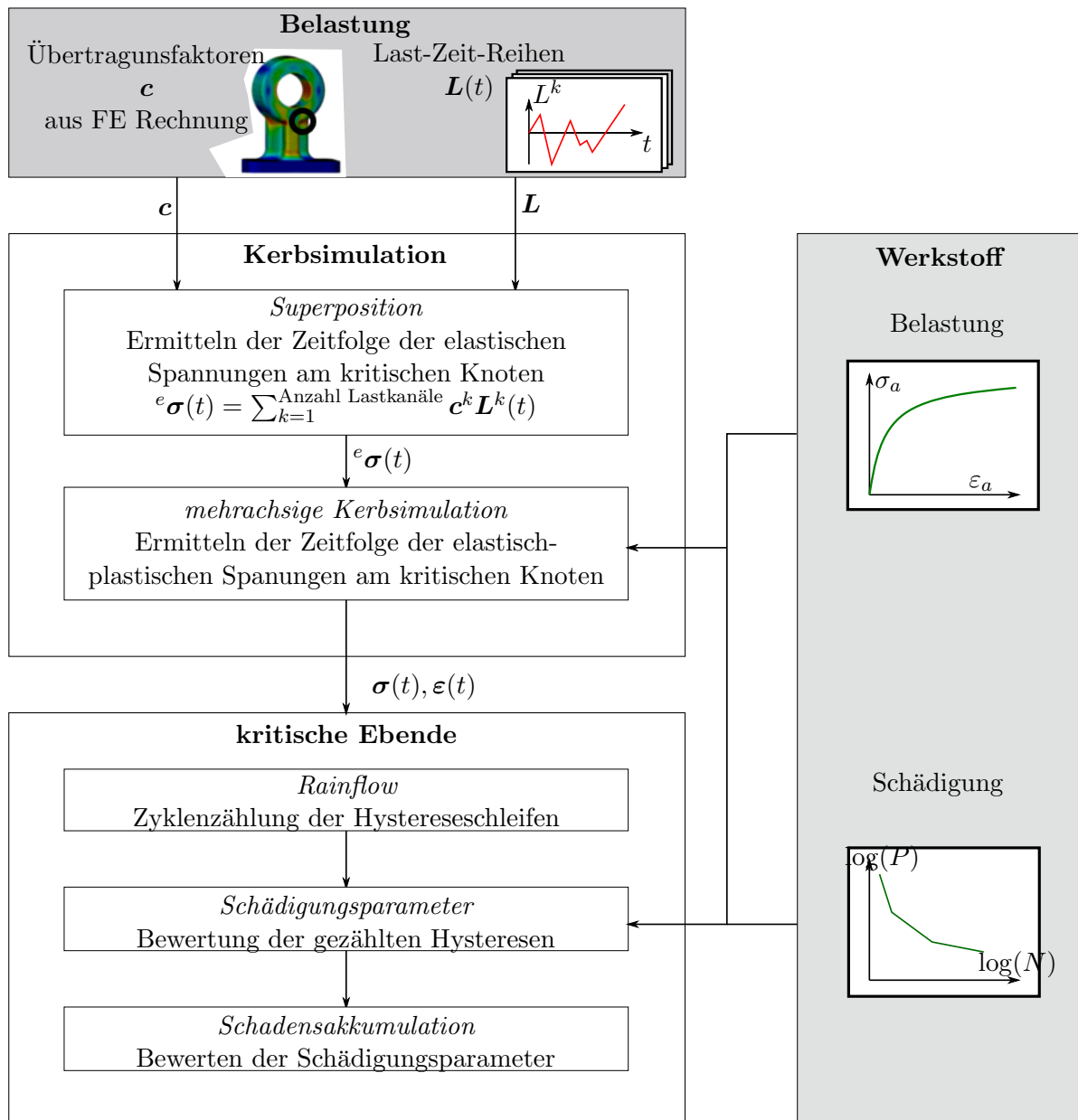


Abbildung 1.1: Übersicht des verwendeten Berechnungsalgorithmus

Zum Verwalten der verschiedenen Optionen und Verfahren ist das Programm objektorientiert aufgebaut. Die zwei großen Programmpakte sind zum einen die Kerbsimulation (mit der Klasse *Kerb-simulation*) und zum anderen die kritischen Ebenen Modelle (Für jedes Modell ist eine Klasse definiert z.B. *PRAM*).

minimaler Input:

- Zugfestigkeit R_m
- Lastfolgen aller Lastkanäle
- Übertragungsfaktoren c

zusätzlicher Input:

- plastische Traglastformzahl K_p
- Spannungsgradient G
- Hochbeanspruchte Oberfläche A_σ

abschätzbarer Input:

- statische Kennwerte (Elastizitätsmodul E und Querdehnzahl ν)
- zyklische Kennwerte (Verfestigungskoeffizient K^{Prime} und -exponent n')
- Schädigungsparameter Wöhlerlinien
- nichtproportionales Verfestigungsvermögen α

Kapitel 2 Lebensdauerrechnung

Die Funktion zum berechnen der Ermüdungslebensdauer heißt *schadensrechnung_kerb.m*. Zum berechnen der Lebensdauer muss eine Instanz der Klasse *Kerbsimulation* definiert sein. Es können beliebig viele Instanzen der einzelnen Schädigungsparameter definiert werden. Zusätzlich muss noch angegeben werden in welchen Schnittebenen die Lebensdauer berechnet werden soll. Die Definition der Schnittebenen erfolgt über zwei Winkel, diese werden im Folgenden erklärt.

Funktionsaufruf

Zum berechnen der Anrisslebensdauer wird die Funktion *schadensrechnung_kerb* verwendet. Definiert sein müssen:

- Schnittwinkel für kritische ebenen Rechnung, siehe Abschnitt 2.1
- Instanz der Klasse *Kerbsimulation*, siehe Kapitel 3
- Instanzen der Klassen der einzelnen Schädigungsparameter, siehe Kapitel 4

```
1 % Winkeldefinition kritische Ebenen
2 % winkel = [phi_max, phi_min, dphi, psi_max, psi_min, dpsi]
3 winkel = [90 0 9 45 0 45];
4 % Alle definierten Schaedigungsparameter als cell array
5 % zusammengefasst
6 DMGs = {Pram, Pfs, Pz};
7 % Rechnung
8 DL = schadensrechnung_kerb(...
9 jobname, outpath, ...      % Name der Rechnung
10 K, ...                    % Instanz der Klasse Kerbsimulation
11 DMGs, ...                 % cellarray aller Schaedigungsparametern
12 winkel, ...               % Winkel der kritischen Ebenen
13 optdisplay, ...           % Display Ausgabe
14 optcritplane, ...         % kritische Ebenen Rechnung
15 opthcm, ...               % Ergebnisse Rainflow in kritischen Ebene
16 optallhcm);               % Ergebnisse Rainflow in allen Ebenen
```

2.1 Definition der Schnittwinkel

Es werden drei Koordinatensystem definiert.

- globales Koordinatensystem $K_g = \{X, Y, Z\}$
- lokales (Kerb)Koordinatensystem $K_l = \{x, y, z\}$, siehe Abbildung 3.1

- lokales (Schnitt)Koordinatensystem $K_s = \{x_s, y_s, z_s\}$, siehe Abbildung 2.1

Im Koordinatensystem K_l wird die Kerbsimulation ausgeführt. Dann hat man die Spannungen $\sigma(t) = [\sigma_{xx}(t) \sigma_{yy}(t) \sigma_{xy}(t)]^T$ und Dehnungen $\varepsilon(t) = [\varepsilon_{xx}(t) \varepsilon_{yy}(t) \varepsilon_{zz}(t) \gamma_{xy}(t)]^T$.

Die kritische Schnittebene wird nach der Kerbsimulation in K_l gesucht. Dazu wird ein drittes Koordinatensystem $K_s = \{x_s, y_s, z_s\}$ eingeführt. Dieses Koordinatensystem wird durch Rotation des lokalen Koordinatensystems K_l eingeführt und beschreibt den aktuellen Schnitt. Die beiden Drehwinkel zur Definition von K_s sind::

- φ , Drehung um die z -Achse. Beschreibt die Drehung der Schnittebene die von der Bauteiloberfläche aus beobachtet werden kann.
- ψ , Drehung um die y_s -Achse. Beschreibt die Drehung der Schnittebene in das Bauteil hinein.

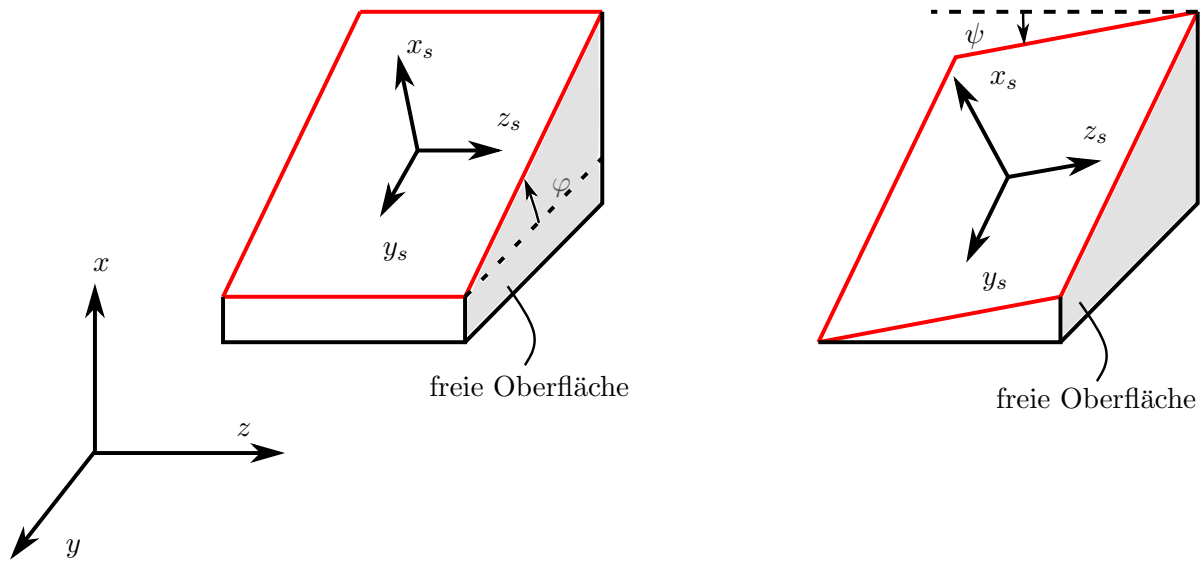


Abbildung 2.1: Lage der Schnittebene im lokalen Koordinatensystem

Die x_s -Achse steht dabei immer senkrecht auf der Schnittebene. In K_s entsteht dann im Allgemeinen ein 3D Spannungszustand. Die aktuelle Schnittebene lässt sich in K_s als Schnitt durch die $y_s - z_s$ Ebene interpretieren. Die Schnittspannung \vec{t} kann also direkt den Komponenten des gedrehten Tensors entnommen werden. Aufgrund der Symmetrie der Schnitte reicht es die Suche auf die Winkel $\varphi \in [0, \pi]$ und $\psi \in [0, \frac{\pi}{2}]$ einzuschränken. Aufgrund der implementierten Parameter wird folgendes empfohlen:

- $\varphi_{min} = 0^\circ$, $\varphi_{max} = 90^\circ$, $\Delta\varphi = 9^\circ$
- $\psi_{min} = 0^\circ$, $\psi_{max} = 45^\circ$, $\Delta\psi = 45^\circ$

Programmausgaben

Als Ergebnis der Rechnung können mehrere Ergebnisdateien sowie eine Displayausgabe erzeugt werden. Die werden über die letzten 4 Inputs der Funktion `schadensrechnung_kerb` gesteuert, `false` = keine Ausgabe `true` = Ausgabe. Die Ergebnisse der Kerbsimulation werden immer in Dateien geschrieben.

- *optdisplay*: Displayausgabe von Zwischenergebnisse und Ergebnissen
- *optcritplane*: Ausgabe der Ergebnisse der kritischen Ebenen Rechnung als Textdatei
- *opthcm*: Ausgabe der Ergebnisse der Rainflowzählung in der kritischen Ebene als Textdatei
- *optallhcm*: Ausgabe der Ergebnisse der Rainflowzählung in allen Ebenen als Textdatei

Ergebnisse Kerbsimulation

Es werden zwei Dateien erzeugt. Beide werden als binary files gespeichert.

- *jobname.pth*: Verlauf der elastischen Kerbspannungen
- *jobname.sig*: Der Verlauf der elastisch-plastischen Kerbspannungen und -dehnungen

Beide können mit der Funktion *fread.m* in Matlab gelesen werden. Zum Lesen der **.sig* Dateien kann auch die Funktion *read_sigepsfile.m* verwendet werden.

Ergebnisse kritische Ebenen, `optcritplane = true`

Die Ergebnisse der kritische Ebenen Rechnung werden in *jobname.cpl* als Text gespeichert. Gelesen werden können die Ergebnisse mit der Funktion *read_CRITPLANE.m*.

Ergebnisse Rainflow Zählung, `opthcm = true`

Die berechneten Schädigungsparameter für jedes Schwingspiel werden in Dateien *jobname_NameParameter_phi_psi.hcm* gespeichert. Die Ergebnisse können für jede Ebene oder nur in der kritischen Ebene ausgegeben werden.

Kapitel 3 Kerbsimulation

In diesem Programmpaket werden die lokalen elastisch-plastischen Spannungen und Dehnungen in der Kerbe ermittelt. Zuerst wird die linear elastischen Spannungen aus den Übertragungsfaktoren und den dazu gehörigen Lasten ermittelt. Anschließend werden diese mit einem mehrachsigen, inkrementellen Plastizitätsmodell in lokale elastisch-plastische Spannungen und Dehnungen überführt. Zum Verwalten der Eingabedaten und Optionen ist die Klasse *Kerbsimulation* implementiert. Eine Instanz der Klasse *Kerbsimulation*:

```
1 % Definition der Klasse Kerbsimulation
2 K = Kerbsimulation(jobname, outpath, L, c, ndl, Rm, E, nu, Kstrich, nstrich, ...
3     'VariablenName', VariablenWert);
```

Dabei sind einige Eingabewerte zwingend erforderlich. Diese sind

Tabelle 3.1: Übersicht erforderlicher Input der Kerbsimulation

Name	Beschreibung	typ	Dimension	Optionen
jobname	Name der Rechnung	string	-	-
outpath	Name des Ordners für Outputdateien	string	-	-
L	Lastfolge	double array	$N_k \times N_D$	-
c	Übertragungsfaktoren	double array	$3 \times N_k$	-
ndl	Anzahl Durchläufe durch L	int > 0	1	-
<i>E</i>	E-Modul	double	1	-
ν	Querdehnzahl	double	1	-
R_m	Zugfestigkeit	double	1	-
K'	Parameter Ramberg Osgood	double	1	-
n'	Parameter Ramberg Osgood	double	1	-

Hier stehen N_k für die Anzahl der am Bauteil angreifenden Lastkanäle und N_d für die Anzahl an Datenpunkten in der Lastfolge. Es wird empfohlen zwischen zwei Umkehrpunkten min. 20 Datenpunkte anzugeben. Die Lastfolge **L** sollte einen Durchlauf enthalten. Insgesamt werden *ndl*

Durchläufe durch die Lastfolge simuliert. Für ndl wird folgendes empfohlen:

- konstante Amplituden ohne Mittellast $ndl = 10$
- konstante Amplituden mit Mittellast $ndl = 1000$
- variable Amplituden $ndl = 2$

Lastfolge

Am Bauteil greifen N_K Lastkanäle an. Für jeden Kanal muss die Lastfolge $L^k(t)$ für diskrete Zeitpunkte $t_0 \dots t_{N_D}$ gegeben sein. Die einzelnen Lastkanäle werden als Zeilenvektoren gespeichert und zur gesamten Lastfolge zusammengefügt.

$$\mathbf{L} \in \mathbb{R}^{N_K \times N_D} \quad \mathbf{L} = \begin{bmatrix} L^1; \dots; L^k; L^{N_K} \end{bmatrix}$$

Bei der Kerbsimulation werden dann ndl Durchläufe durch die Lastfolge berechnet.

Übertragungsfaktoren

Um von den äußeren Lasten auf die Lokale Beanspruchung schließen zu können werden Übertragungsfaktoren benötigt. Dazu können z.B. linear elastische FE Rechnungen mit Einheitslasten durchgeführt werden. Dabei sollte ein lokales Kerbkoordinatensystem gewählt werden, sodass sich an der Bauteiloberfläche ein ESZ einstellt (eine Achse wird als Normale an die Bauteiloberfläche gewählt). Ein Beispiel ist dargestellt in Abbildung 3.1.

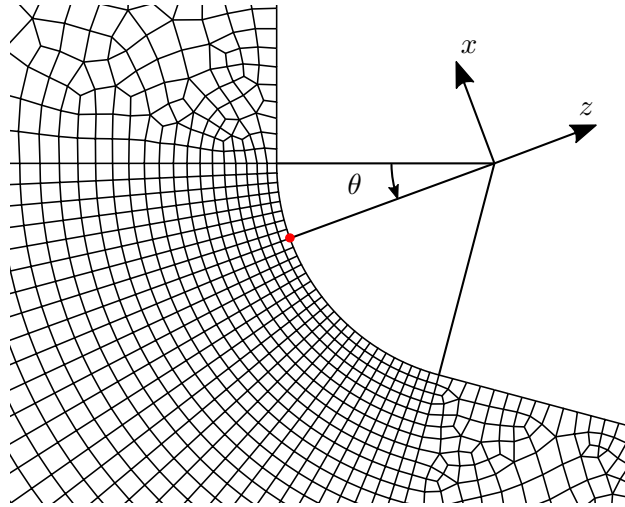


Abbildung 3.1: Lokales Koordinatensystem in der Kerbe

Für jeden Lastkanal L^k die Übertragungsfaktoren $c^k = [c_{xx}^k, c_{yy}^k, c_{xy}^k]^T$ bestimmt werden. Diese werden ebenfalls in einer Matrix gespeichert.

$$\mathbf{c} \in \mathbb{R}^{3 \times N_K} \quad \mathbf{c} = \begin{bmatrix} c^1, \dots, c^k, \dots, c^{N_K} \end{bmatrix}$$

Die Übertragungsfaktoren können dabei aus einer statischen linear elastischen FE Rechnungen gewonnen werden. Dabei werden die lokalen (elastizitätstheoretischen) Spannungen ${}^e\hat{\sigma}^k$ auf eine festgelegte Last \hat{L} bezogen.

$$c_{ij}^k = \frac{{}^e\hat{\sigma}_{ij}^k}{\hat{L}^k}$$

Die lokalen Spannungen ergeben sich dann aus der Summe über die einzelnen Lastkanäle.

$${}^e\boldsymbol{\sigma} = \sum_{k=1}^{N_k} c^k L^k = \mathbf{c} \mathbf{L} \in \mathbb{R}^{3 \times N_D} \quad (3.1)$$

optionaler Input

Darüber hinaus können einige Optionen verändert werden. Dazu muss zuerst der Name der Variable als String und anschließend der Wert der Variable übergeben werden.

Tabelle 3.2: Übersicht des optionalen Inputs der Kerbsimulation

Name	Beschreibung	typ	Dimension	Optionen
material	Auswahl des Materialmodells	string	-	Chaboche KarimOhno OhnoWang(default) PseudoStress(default)
verfahren	Name des Kerb- näherungsverfahrens	string	-	PseudoStrain DevEps Neuber(default)
eindkerb	Option zum bestimmen der Bauteilfließkurve	string	-	NeuberStern SeegerBeste
nppara	Wahl des Nicht- proportionalitätsfaktors	string	-	None(default) Riess Meggiolaro Gaier
α	Materialparameter nichtpro- portionale Verfestigung	double	-	-
K_p	Traglastformzahl	double	-	-

Soll zum Beispiel die nichtproportionale Verfestigung berücksichtigt werden, so muss die Variable "nppara" gesetzt werden.

```

1 % Definition der Klasse Kerbsimulation mit NP Verfestigung
2 K = Kerbsimulation(jobname,outputpath,L,c,ndl,Rm,E,nu,Kstrich,nstrich,...
3 'nppara','Gaier','alpha',0.3);

```

Die Angabe von α ist dabei optional, wird nichts angegeben wird α abgeschätzt.

Kapitel 4 Schädigungsparameter

Zur Berechnung der Anrisslebensdauer sind drei verschiedene Modell implementiert. Jedes Modell ist dabei in jeweils in zwei Varianten implementiert. In der ersten Variante werden Schädigungsparameter Wöhlerlinien aus den Parametern der Dehnungswöhlerlinie berechnet, in der zweiten Variante müssen Steigungen und Stützpunkte der Schädigungsparameter Wöhlerlinien angegeben werden. Die implementierten Parameter sind:

1. Smith-Watson-Topper Modell: P_{RAM} , $P_{RAM,stuetz}$
2. Fatemi-Socie Modell: P_{FS} , $P_{FS,stuetz}$
3. Kurzrissmodell: P_Z , $P_{Z,stuetz}$

4.1 Smith-Watson-Topper P_{RAM}

```
1 % PRAM definiert durch Dehnungswöhlerlinie
2 Pram = PRAM(E, sf, ef, b, c, ND, Msig, 'VariablenName', VariablenWert);
3 % PRAM definiert durch Stuetzpunkte der P-Wöhlerlinie
4 Pram = PRAM_stuetz(E, Pram_WS_stuetz, d1, d2, Msig, fak, ...
5                     'VariablenName', VariablenWert);
```

Tabelle 4.1: Übersicht Input des Parameters *PRAM*

Name	Beschreibung	typ	Dimension	Optionen
E	E-Modul	double	1	-
σ'_f	Parameter DehnungsWL	double	1	-
ε'_f	Parameter DehnungsWL	double	1	-
b	Parameter DehnungsWL	double	1	-
c	Parameter DehnungsWL	double	1	-
ND	Dauerfestigkeit	double	1	-
M_σ	Mittelspannungsepfndlichkeit	double	1	-
nst	Stützziffer	double	1	optional
miner	Option Miner Regel	double	1	optional 0 elementar (default) 2 original

Tabelle 4.2: Übersicht Input des Parameters $PRAM_stuetz$

Name	Beschreibung	typ	Dimension	Optionen
E	E-Modul	double	1	-
$P_{ram,WS,stuetz}$	Stützstelle P-Wöhlerlinie Werkstoff	double	1	-
N_stuetz	Stützstelle in N zu $P_{RAM,WS,stuetz}$	double	1	optional default = 1000
$P_ram_WSD_stuetz$	Stützstelle PWL Werkstoff Dauerfestigkeit	double	1	optional default = $1e - 20$
d_1	Neigung P-Wöhlerlinie	double	1	-
d_2	Neigung P-Wöhlerlinie	double	1	-
M_σ	Mittelspannungsepfindlichkeit	double	1	-
fak	Faktor zum Übergang von Werkstoff auf Bauteil WL wie in Richtlinie Nichtlinear	double	1	-
miner	Option Miner Regel	double	1	optional 0 elementar (default) 2 original

4.2 Fatemi-Socie P_{FS}

```

1 % Fatemi-Socie definiert durch Dehnungswöhlerlinie
2 Pfs = PFS(E,nu,sigF,sf,ef,b,c,ND,'VariablenName',VariablenWert);
3 % Fatemi-Socie definiert durch Stuetzpunkte
4 Pfs = PFS_stuetz(E,nu,kfs,Pfs_WS_stuetz,d1,d2,fak,...
5     'VariablenName',VariablenWert);

```

Tabelle 4.3: Übersicht Input des Parameters *PFS*

Name	Beschreibung	typ	Dimension	Optionen
E	E-Modul	double	1	-
ν	Querdehnzahl	double	1	-
σ_F	Fließspannung	double	1	-
σ'_f	Parameter DehnungsWL	double	1	-
ε'_f	Parameter DehnungsWL	double	1	-
b	Parameter DehnungsWL	double	1	-
c	Parameter DehnungsWL	double	1	-
ND	Dauerfestigkeit	double	1	-
nst	Stützziffer	double	1	optional
miner	Option Miner Regel	double	1	optional 0 elementar (default) 2 original
fwt	Schubwechselfestigkeit	double	1	optional (default = $1/\sqrt{3}$)
kfs	Parameter/Fließspannung	double	1	optional wird automatisch bestimmt

Tabelle 4.4: Übersicht Input des Parameters PFS_stuetz

Name	Beschreibung	typ	Dimension	Optionen
E	E-Modul	double	1	-
ν	Querdehnzahl	double	1	-
k_{FS}	Parameter/Fließspannung	double	1	-
$P_{fs,WS,stuetz}$	Stützstelle P-Wöhlerlinie Werkstoff	double	1	-
N_stuetz	Stützstelle in N zu $P_{FS,WS,stuetz}$	double	1	optional default = 1000
$P_fs_WSD_stuetz$	Stützstelle PWL Werkstoff Dauerfestigkeit	double	1	optional default = $1e - 20$
d_1	Neigung P-Wöhlerlinie	double	1	-
d_2	Neigung P-Wöhlerlinie	double	1	-
fak	Faktor zum Übergang von Werkstoff auf Bauteil WL wie in Richtlinie Nichtlinear	double	1	-
miner	Option Miner Regel	double	1	optional 0 elementar (default) 2 original

4.3 Kurzrissmodell P_Z

```

1 % Kurzrissmodell definiert durch Dehnungswöhlerlinie
2 Pz = PZ2(E,nu,Rm,Kstrich , nstrich , sf , ef , b , c , ND,...
3         'VariablenName', VariablenWert );
4 % Kurzrissmodell definiert durch Stuetzpunkte
5 Pz = PZ2_stuetz(E,nu , sigF , Pz_WS_stuetz , d , fak ,
6         'VariablenName', VariablenWert );

```

Tabelle 4.5: Übersicht Input des Parameters *PZ2*

Name	Beschreibung	typ	Dimension	Optionen
E	E-Modul	double	1	-
ν	Querdehnzahl	double	1	-
R_m	Zugfestigkeit	double	1	-
K'	Parameter Ramberg Osgood	double	1	-
n'	Parameter Ramberg Osgood	double	1	-
σ'_f	Parameter DehnungsWL	double	1	-
ε'_f	Parameter DehnungsWL	double	1	-
b	Parameter DehnungsWL	double	1	-
c	Parameter DehnungsWL	double	1	-
ND	Dauerfestigkeit	double	1	-
nst	Stützziffer	double	1	optional
fwf	Schubwechselfestigkeit	double	1	optional (default = $1/\sqrt{3}$)
sigF	Fließspannung	double	1	optional
tauF	Fließspannung Torsion	double	1	optional
ce	(halbe) Endrisslänge an der Oberfläche	double	1	optional default = 0.25mm
M_sig	Mittelspannungsepfndlichkeit	double	1	optional
Gsig	Spannungsgradient	double	1	optional default = 0 1/mm
Gtau	Spannungsgradient	double	1	optional default = 0 1/mm
FA	Materialparameter	double	1	optional default = 1000
mu	Materialparameter	double	1	optional default = 0.5
takt	Materialparameter	double	1	optional default = 50 MPa
ators	Materialparameter	double	1	optional default = 0

Tabelle 4.6: Übersicht Input des Parameters *PZ2_stuetz*

Name	Beschreibung	typ	Dimension	Optionen
E	E-Modul	double	1	-
ν	Querdehnzahl	double	1	-
sigF	Fließspannung	double	1	-
$P_{z,WS,stuetz}$	Stützstelle P - Wöhlerlinie	double	1	-
N_stuetz	Stützstelle in N zu $W_{eff,WS,stuetz}$	double	1	optional default = 1
Pz_WSD_stuetz	Stützstelle P - Wöhlerlinie Werkstoff Dauerfestigkeit	double	1	optional default = NaN
ND_stuetz	Lebensdauer Werkstoff Dauerfestigkeit	double	1	optional default = 1e6
d	Neigung P-Wöhlerlinie	double	1	-
fak	Faktor zum Übergang von Werkstoff auf Bauteil WL wie in Richtlinie Nichtlinear	double	1	-
fwt	Schubwechselfestigkeit	double	1	optional (default = $1/\sqrt{3}$)
tauF	Fließspannung Torsion	double	1	optional
ce	(halbe) Endrisslänge an der Oberfläche	double	1	optional default = 0.25mm
M_sig	Mittelspannungsepfindlichkeit	double	1	optional
Gsig	Spannungsgradient	double	1	optional default = 0 1/mm
Gtau	Spannungsgradient	double	1	optional default = 0 1/mm
FA	Materialparameter	double	1	optional default = 1000
mu	Materialparameter	double	1	optional default = 0.5
takt	Materialparameter	double	1	optional default = 50 MPa
ators	Materialparameter	double	1	optional default = 0

Kapitel 5 Abschätzen von Kennwerten

Sollten Materialdaten nicht vorliegen können diese auch anhand der Zugfestigkeit R_m und der Werkstoffgruppe abgeschätzt werden. Dabei sind die drei Werkstoffgruppen Stahl, Stahlguss, Aluminiumknet. aus der Richtlinie Nichtlinear enthalten. Sofern vorhanden ist auch eine Abschätzung der neuen Werkstoffgruppe höchstfester Stahl, siehe Abschlussbericht "Höchstfeste Stähle", Vorhaben Nr. 610 2021, möglich.

statische Kennwerte

```
1 % Abschaetzen statischer Kennwerte
2 [E,nu] = StatischFKM('Stahl');
```

zyklische Kennwerte

```
1 % Abschaetzen zyklischer Kennwerte
2 [Kstrich,nstrich] = RambergOsgoodFKM('Stahl',Rm);
```

Stützziffer

```
1 % Stuetzziffer
2 % Asig – Hochbeanspruchte Oberflaeche
3 % G – bezogener Spannungsgradient 1/mm
4 % n – gesamte Stuetzziffer
5 % nst – statistische Stuetzziffer
6 % nbm – bruchmechanische Stuetzziffer
7 [n,nst,nbm] = StuetzZifferFKM(Asig,G,Rm,'Stahl');
```

Mittelspannungsempfindlichkeit

```
1 % Abschaetzen Mittelspannungsempfindlichkeit
2 Msig = MittelspannungseinflussFKM('Stahl',Rm);
```

Schädigungsparameter Wöhlerlinien

Die Schädigungsparameter Wöhlerlinien für P_{RAM} kann aus der Richtlinie Nichtlinear entnommen werden. Die Wöhlerlinien für P_Z kann VORLÄUFIG aus der Wöhlerlinien für P_{RAJ} abgeschätzt werden. Für den Fatemi-Socie Parameter liegt derzeit noch keine zuverlässige Abschätzmethode vor, die angegebene Funktion ist eher als Platzhalter zu sehen.

```

1 % PRAM
2 [ Pram_WS_stuetz , Pram_WSD_stuetz , d1 , d2 , f0025 ] = ...
3                                     Pram_Woehlerlinie_FKM (Rm, 'Stahl' );
4 % PZ ( vorlaeufig )
5 [ Pz_WS_stuetz , Pz_WSD_stuetz , d_z , f0025 ] = ...
6                                     Pz_Woehlerlinie (Rm, wsgruppe );
7 % PFS ( vorlaeufig )
8 [ Pfs_WS_stuetz , Pfs_WSD_stuetz , d1 , d2 ] = ...
9                                     Pfs_Woehlerlinie (Rm, 'Stahl' );

```

Rauheitsfaktor

```

1 % Rauheitsfaktor
2 % Rz      – Oberflaechenrauheit in mikrometer
3 % Rm      – Zugfestigkeit
4 KPR = RauheitseinflussFKM ( wsgruppe , Rz , Rm );

```

Absicherung der Bauteilwöhlerlinien

```

1 % P_A      – Ausfallwahrscheinlichkeit
2 % n        – Stuetzziffer
3 % KPR      – Rauheitsfaktor
4 % f0025    – Faktor zum Absichern der P-woehlerlinie
5 %          des Werkstoffs
6 % Absicherung Bauteilwoehlerlinie Pram
7 fram = AbsichernBauteilWoehlerlinie_Pram (P_A , n , KPR , f0025 );
8 % Absicherung Bauteilwoehlerlinie Pz
9 fraj = AbsichernBauteilWoehlerlinie_Praj (P_A , n , KPR , f0025 );

```

Absicherung der Lastfolge

```

1 % Absicherung Lastfolge
2 % verfahren – Verfahren zum absichern (1 2 3)
3 % s_L      – Streuung Lmax
4 % Lmax     – maximalwert Lastfolge
5 % P_L      – Auftretenswahrscheinlichkeit der Lastfolge
6 % P_A      – Ausfallwahrscheinlichkeit
7 gamma_L1 = AbsicherungLastfolgeFKM ( verfahren , s_L , Lmax , P_L , P_A );

```

Kapitel 6 Beispielrechnung

Als Beispiel wird eine Versuchsreihe aus dem Projekt "Mehrachsig Örtlich" berechnet. Berechnet werden die Versuche unter 90° Phasenverschiebung für den Baustahl S355 und den Kerbstab mit Kerbradius $\rho = 0.5$ mm. Das Berechnungsskript liegt im Ordner "06_README" unter dem Namen *BEISPIEL_K05_NP90.m*. Nachfolgend ist das Berechnungsskript angegeben. Als Ergebnis erhält man die Bauteilwöhlerlinie für die drei implementierten Schädigungsparameter, für beide implementierten Varianten.

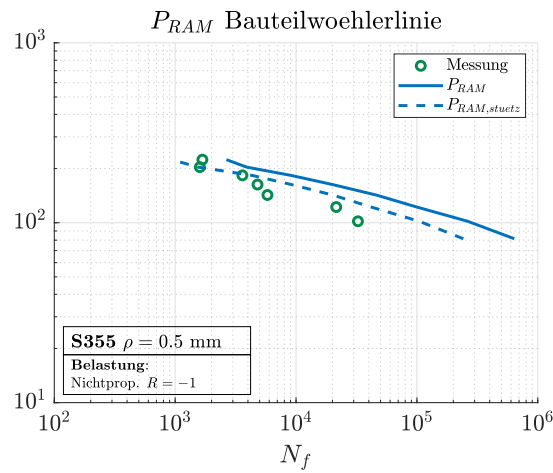


Abbildung 6.1: Bauteilwöhlerlinie P_{RAM}

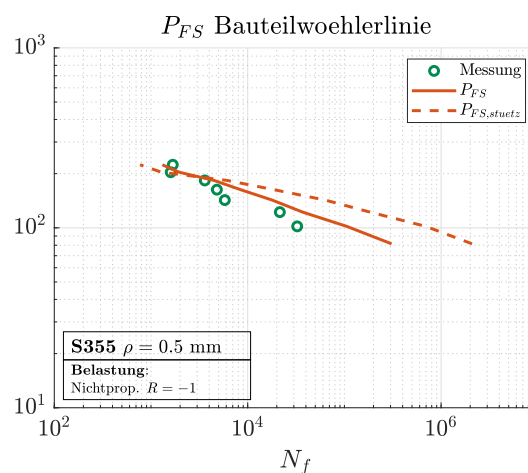


Abbildung 6.2: Bauteilwöhlerlinie P_{FS}

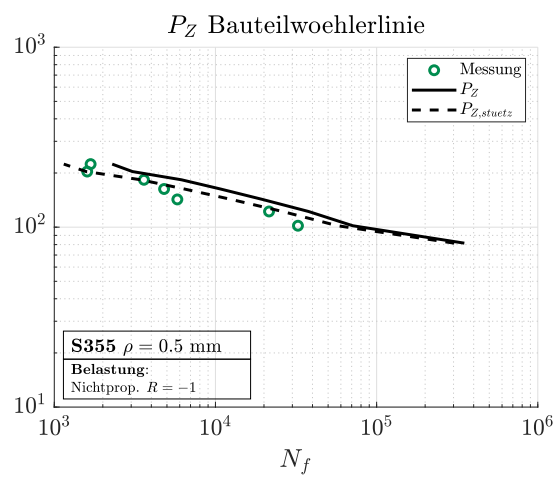


Abbildung 6.3: Bauteilwöhlerlinie P_Z

Table of Contents

.....	1
Definitionen Rechnung	1
Definition Materialmodell	1
Definitionen Bauteil	2
Absicherung	2
Definition Kerbsimulation	2
Definition Lastfolgen	2
Abschaetzen Kennwerte nach FKM	3
Rechnung	3
Auswerten der Rechnung	11

```
% #####
% # Beispiel: Berechnen einer Bauteilfließkurve #
% # # #
% # Werkstoff: Baustahl S355 #
% # Bauteil : Abgesetzte Welle, Kerbradius 0.5mm #
% # Belastung: 90° Phasenverschiebung, R = -1 #
% # Lastkanal 1 - Zug/Druck #
% # Lastkanal 2 - Torsion #
% #####

% Clear Matlab
clear
clc
% Füge alle Funktionen Matlab Pfad hinzu
% run ../startup
```

Definitionen Rechnung

```

outputpath = '../00_Temp';           % Name des Output Folders
jobpart = 'K05_NP_';                % Name der Rechnung

if ~exist(outputpath,'dir')
    mkdir(outputpath)
end

```

Definition Materialmodell

```
E = 204000; % E-Modul
nu = 0.3; % Querdehnzahl
Rm = 541; % Zugfestigkeit
% zyklisch stabil (Ramberg Osgood)
Ks = 839;% 1141;% % zyklische Steifigkeit
ns = 0.138;% 0.193;% % verfestigungsexponent
% Dehnungswoeehlerlinie
sf = 762; ef = 0.415; b = -0.08; c = -0.556;
ND=5e5;
```

Definitionen Bauteil

```
% Übertragungsfaktoren Kerbradius 0.5mm
%      xx      yy      yy
cf = [3.351 0.8351 0.0;...           % Nettospannung Zug
      0.0   0.0   1.963]';         % Nettospannung Torsion
Kp = 2.7;                           % Traglastformzahl
Asig = 25;                          % Hochbeanspruchte Oberfläche
Gsig = 2/0.5;                       % Gradient Normalspannung
Gtau = 1/0.5 + 2/25;                % Gradient Schubspannung
Gmises = sqrt(Gsig^2+3*Gtau^2);
Rz = 0;                             % Rauheit (unberücksichtigt)
```

Absicherung

```
% Ausfallwahrscheinlichkeit (Tabelle 2.4 Richtlinie Nichtlinear)
P_A = 0.5;
% Auftretenswahrscheinlichkeit der Lastfolge
P_L = 0.5;
% Standardabweichung Absicherung der Lastfolge
s_L = 0;
```

Definition Kerbsimulation

```
% Bestimmen der Bauteilfließkurve
eindkerb = 'Seeger Beste';
% Nichtproportionale Verfestigung
nppara = 'Gaier';
% Anzahl der zu simulierenden Durchläufe durch die Lastfolge
ndl = 10;
```

Definition Lastfolgen

```
% Anzahl Datenpunkte pro Zyklus
ndp = 64;
% Lastfolge der Nettospannung Zug mit Amplitude 1
LZ = lastgenerator(1,1,90,1,0,ndp); % = cos(2*pi*t);
% Lastfolge der Nettospannung Torsion mit Amplitude 1
LT = lastgenerator(1,1,0,1,0,ndp); % = sin(2*pi*t);
% Zusammenfassung
L = [LZ;LT];
% Amplituden der Nettospannungen aus den Versuchen
%      Zug/Druck      Torsion
AMP = [ ...
      203.7183272 181.0829575;...
      142.602829 126.7580702;...
      81.48733086 72.43318299;...
      101.8591636 90.54147874;...
      122.2309963 108.6497745;...
      162.9746617 144.866366;...
      224.0901599 199.1912532;...
      183.3464944 162.9746617...
```

```
];
```

Abschaetzen Kennwerte nach FKM

```
% Werkstoffgruppe
wsgruppe = 'Stahl';
% Rauheitsfaktor
KPR = RauheitseinflussFKM(wsgruppe,Rz,Rm);
% Stuetzwirkung Richlinie Nichtlinear
[n,nst,nbm] = StuetzZifferFKM(Asig,Gmises,Rm,wsgruppe);
% Pram - Woehlerlinie
Msig = MittelspannungseinflussFKM(wsgruppe,Rm);
[Pram_WS_stuetz,Pram_WSD_stuetz,d1_ram,d2_ram,f0025] = ...
    Pram_Woehlerlinie_FKM(Rm,wsgruppe);
% Absicherung Bauteilfließkurve
fram = AbsichernBauteilWoehlerlinie_Pram(P_A,n,KPR,f0025);
% Pfs - Woehlerlinie (vorlaufig)
[Pfs_WS_stuetz,Pfs_WSD_stuetz,d1_fs,d2_fs] = ...
    Pfs_Woehlerlinie(Rm,wsgruppe);
% Pz - Woehlerlinie (vorlaufig aus Praj Woehlerlinie)
[Pz_WS_stuetz,Pz_WSD_stuetz,d_z,f0025] = ...
    Pz_Woehlerlinie(Rm,wsgruppe);
% Absicherung Bauteilfließkurve
fraj = AbsichernBauteilWoehlerlinie_Praj(P_A,n,KPR,f0025);
```

Rechnung

```
% Zum berechnen der Bauteilwöhlerlinie werden alle Amplituden aus der
% Versuchsreihe nachgerechnet. Dazu wird in jedem Schleifendurchlauf die
% Lastfolge skaliert. Da alle Rechnungen unabhängig von einander sind,
% können diese auch parallel ausgeführt werden.

% parfor i = 1:size(AMP,2) % Parallele Rechnung, ! optdisplay auf 0 setzten

for i = 1:size(AMP,2) % Sequenzielle Rechnung
    % Amplitude aktueller Versuch
    amp = AMP(:,i);
    last = amp .* L;
    % Absicherung Lastfolge
    gamma_L1 = AbsicherungLastfolgeFKM(1,s_L,max(last(1,:)),P_L,P_A);
    gamma_L2 = AbsicherungLastfolgeFKM(1,s_L,max(last(2,:)),P_L,P_A);
    last = [gamma_L1; gamma_L2] .* last;
    % Jobname
    jobname = [jobpart,num2str(amp(1))];
    % Definition einer Instanz der Kerbsimulation
    K = Kerbsimulation(jobname,outpath,...
        last,cf,ndl,Rm,E,nu,Ks,ns,...
        'eindkerb',eindkerb,...
        'nppara',nppara,...
        'Kp',Kp);
    % Definition einer Instanz des Parameters PRAM
    % 1. Pram aus DehnungsWL
    % 2. Pram anhand abgeschaetzter Stuetzstellen
```

```

Pram = PRAM(E,sf,ef,b,c,ND,Msig,'nst',n);
Pram_s = PRAM_stuetz(E,Pram_WS_stuetz,dl_ram,d2_ram,Msig,fram);
% Definition einer Instanz des Pramaters PFS
% 1. Pfs aus DehnungswL
% 2. Pfs anhand abgeschaeetzter Stuetzstellen
Rp02s = Ks*0.002^ns;
sigF = 0.5 *(Rm+Rp02s); % Fließgrenze als Mittelwert aus Rp02 und Rm
Pfs = PFS(E,nu,sigF,sf,ef,b,c,ND,'nst',n);
Pfs_s = PFS_stuetz(E,nu,Pfs.kfs,Pfs_WS_stuetz,dl_fs,d2_fs,fram);
% Definition einer Instanz des Pramaters PZ
% 1. Pz aus DehnungswL
% 2. Pz anhand abgeschaeetzter Stuetzstellen
Pz = PZ2(E,nu,Rm,Ks,ns,sf,ef,b,c,ND,'nst',nst,'Gsig',Gsig,'Gtau',Gtau);
Pz_s = PZ2_stuetz(E,nu,sigF,Pz_WS_stuetz,d_z,fracj,...
    'Pz_WSD_stuetz',Pz_WSD_stuetz,'Gsig',Gsig,'Gtau',Gtau);
% Zusammenfassen aller Instanzen der definierten Schaedigungsparameter
% als cell array
%   DMGs = {Pram, Pram_s, Pfs, Pfs_s, Pz, Pz_s};
DMGs = {Pram_s, Pfs_s, Pz_s};
% winkel = [phi_max, phi_min, dphi, psi_max, psi_min, dpsil]
winkel = [90 0 9 45 0 45];
% Funktion zum berechnen der Anrisslebensdauer
[DL,DLc,phic,psic] = schadensrechnung_kerb(...
    jobname,outputpath,... % Name der Rechnung
    K,... % Instanz der Kerbsimulation
    DMGs,... % Cell array Instanzen Schaedigungsparameter
    winkel,... % Winkel kritische Ebenen Rechnung
    1,... % Display ausgabe
    1,... % File Ausgabe Ergebnisse kritische Ebene
    1,... % Ausgabe Ergebnisse Rainflow Zählung
    0); % Rainflow Zählung in allen Ebenen
end

% Ende der Rechnung

*****
*****
Jobname           :K05_NP_203.7183
Verfahren         :PseudoStress Lang
Materialmodell    :OhnoWang
Backstresstensenoren:10
Anzahl Druchläufe :10
Anzahl Inkemente :731
Starte Kerbsimulation, Ende, Dauer: 0.14563
-----
Starte kritische Ebenen Rechnung
dphi: 9.000, phimin: 0.000, phimax: 90.000
dpsi: 45.000, psimin: 0.000, psimax: 45.000
Anzahl zu berechnender Ebenen:22
plane phi psi PRAM_st PFS_st PZ_st
1 0.000 0.000 1.550649e+03 1.282646e+03 001578
2 0.000 0.785 7.391041e+05 3.687602e+03 015914
3 0.157 0.000 1.598873e+03 1.420013e+03 1.722342e+03
4 0.157 0.785 7.333245e+05 3.796672e+03 016091

```

5	0.314	0.000	1.668520e+03	1.908694e+03	2.082342e+03
6	0.314	0.785	3.740981e+05	4.126584e+03	021448
7	0.471	0.000	1.847728e+03	2.977884e+03	2.408342e+03
8	0.471	0.785	2.098148e+05	4.920658e+03	025882
9	0.628	0.000	2.294416e+03	4.758497e+03	003069
10	0.628	0.785	1.524697e+05	6.536367e+03	037538
11	0.785	0.000	3.417172e+03	6.419603e+03	004592
12	0.785	0.785	1.448524e+05	1.019922e+04	057022
13	0.942	0.000	6.575445e+03	6.523798e+03	007875
14	0.942	0.785	1.769053e+05	1.965073e+04	086576
15	1.100	0.000	1.788998e+04	5.638953e+03	025007
16	1.100	0.785	2.752081e+05	4.921371e+04	102852
17	1.257	0.000	7.916952e+04	5.064171e+03	015692
18	1.257	0.785	5.500399e+05	1.739992e+05	083855
19	1.414	0.000	5.848793e+05	5.152508e+03	010724
20	1.414	0.785	1.327082e+06	8.228557e+05	055708
21	1.571	0.000	1.752325e+06	5.406506e+03	010408
22	1.571	0.785	2.235824e+06	2.185375e+06	053754

Ende, Dauer krtische Ebene: 0.20242

Dauer gesamt : 0.34805

kritische Ebene:

Parameter	phi	psi	DL
PRAM_st:	0.000	0.000	1550.649
PFS_st:	0.000	0.000	1282.646
PZ_st:	0.000	0.000	1578.000

Jobname :K05_NP_142.6028

Verfahren :PseudoStress Lang

Materialmodell :OhnoWang

Backstresstensoren:10

Anzahl Druchläufe :10

Anzahl Inkemente :731

Starte Kerbsimulation, Ende, Dauer: 0.06268

 Starte kritische Ebenen Rechnung

dphi: 9.000, phimin: 0.000, phimax: 90.000

dpsi: 45.000, psimin: 0.000, psimax: 45.000

Anzahl zu berechnender Ebenen:22

plane	phi	psi	PRAM_st	PFS_st	PZ_st
1	0.000	0.000	2.004938e+04	6.347016e+04	012789
2	0.000	0.785	4.916696e+06	1.680934e+05	1.332272e+05
3	0.157	0.000	2.117762e+04	7.032806e+04	012605
4	0.157	0.785	6.199162e+06	1.731985e+05	127168
5	0.314	0.000	2.269513e+04	9.433685e+04	012528
6	0.314	0.785	4.989459e+06	1.877503e+05	150812
7	0.471	0.000	2.610851e+04	1.467831e+05	014871
8	0.471	0.785	3.581684e+06	2.232811e+05	201723
9	0.628	0.000	3.345289e+04	2.335779e+05	017838
10	0.628	0.785	2.929076e+06	2.945899e+05	329453
11	0.785	0.000	5.165491e+04	3.110619e+05	024664
12	0.785	0.785	2.976485e+06	4.557608e+05	1.000000e+20