

**Univerza v Ljubljani  
Fakulteta za strojništvo**

**Realna kontaktna površina in temperatura  
Poročilo laboratorijske vaje**

Rok oddaje: Petek, 18. 3. 2016

Uroš R

15. junij 2016

## Kazalo

<b>1</b>	<b>Realna kontaktna površina</b>	<b>3</b>
1.1	Uvod . . . . .	3
1.1.1	Metoda upoštevanja porazdelitve hrapavostnega profila . . . . .	3
1.2	Izračun . . . . .	4
1.2.1	Metoda 1 - glavni program in rezultati . . . . .	4
1.2.2	Metoda 2 - glavni program in rezultati . . . . .	10
1.2.3	Funkcija GW_model . . . . .	13
1.3	Zbirnik rezultatov . . . . .	13
1.4	Zaključek - realna kontaktna površina . . . . .	14
<b>2</b>	<b>Realna kontaktna temperatura</b>	<b>15</b>
2.1	Uvod . . . . .	15
2.2	Glavni program . . . . .	15
2.2.1	Metoda 1 . . . . .	15
2.2.2	Metoda 2 . . . . .	19
2.3	Zaključek - temperature v kontaktu . . . . .	21
	<b>Literatura</b>	<b>28</b>

# 1. Realna kontaktna površina

## 1.1 Uvod

Problem kontaktne površine je povezan z dejstvom, da nobena od površin ni absolutno ravna. Izhajajoč iz osnovnih izmerkov hrapavostnih parametrov ter nekaterih predpostavk, je mogoče v okviru statistične analize določiti na kakšen način je povezana realna kontaktna površina s silo obremenjevanja, posledično pa je možno izračunati tudi parametre kot je npr. stični tlak v kontaktu. Kot osnova služi Hertzova teorija kontaktov teles, katera je bila modificirana za parametre na hrapavostni skali. Za izpeljavo modela je potrebno predpostaviti sledeče [1], [2]:

1. Materiali v kontaktu so izotropni in se deformirajo samo elastično,
2. Kontaktna napetost je povzročena zaradi obremenitev normalnih na ravnino na kateri se dve telesi stikata, torej na površini ni nobenih strižnih sil,
3. Kontaktna ploskev je v primerjavi s celotno ploskvijo zelo majhna,
4. Telesa mirujejo,
5. Kontakt med površinama je med absolutno ravno in hrapavo površino,
6. Porazdelitev višine vršičkov je opisana s funkcijo  $\phi(z)$ ,
7. Vsi vršički so sferične oblike in imajo enak radij  $R$ ,
8. Referenčna ravnina ( $z = 0$ ) poteka preko srednje vrednosti višine vršičkov,
9. Vršički z višino višjo od  $d$  bodo formirali kontakt.

Po izpeljavi je možno zapisati, da je funkcija sile ter površine glede na oddaljenost  $d$  sledeča:

$$A(d) = \pi NR \int_d^{\infty} \phi(z)(z - d) dz \quad (1)$$

$$F(d) = \frac{4}{3} NE^* \sqrt{R} \int_d^{\infty} (z - d)^{\frac{3}{2}} \phi(z) dz \quad (2)$$

### 1.1.1 Metoda upoštevanja porazdelitve hrapavostnega profila

Upoštewane sta bile dve metodi:

- $\phi(z)$  je normalna porazdelitev hrapavostnega profila

$$\phi(z) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\left[-\frac{z^2}{2\sigma^2}\right]} \quad (3)$$

- $\phi(z)$  je eksponentna porazdelitev hrapavostnega profila

$$\phi(z) = \frac{1}{\sigma} e^{\left[-\frac{1}{\sigma}z\right]} \cdot H(z) \quad (4)$$

V enačbi 3 je že predpostavljeno, da je porazdelitev simetrična glede na srednjo linijo profila.  $H(z)$  je skočna oz. Heavisidova funkcija.

Enačbi 1 in 2 sta analitično težje rešljivi. V ta namen si pomagamo z programskim orodjem Matlab, kateri preko Simpsonove aproksimacijske formule izračuna integral.

## 1.2 Izračun

Program je strukturiran na dva dela. Prvi del je glavni program, kjer so zapisani podatki ter ukazi za izris grafov, drugi del je funkcija, ki izračuna podatke za te grafe ter poda rezultate.

### 1.2.1 Metoda 1 - glavni program in rezultati

```
clc
clear
close all
format longG
```

#### Vhodni podatki

```
data1.name = 'Imenska hrapavost = 0,6 um';
data1.R = 5.5;%um radius of tops
data1.sigma = 6.6e-1;%um standard roughness deviation
data1.eta = 1.7e10;%[m^-2]number of tops density
data1.An = 4.3e4;%um^2 nominal area
%
data2.name = 'Imenska hrapavost = 0,06 um';
data2.R = 1.7e1;%um radius of tops
data2.sigma = 8.4e-2;%um standard roughness deviation
data2.eta=2.9e10;%[m^-2]number of tops density
data2.An=4.3e4;%um^2 nominal area
%
E1=210e-3;%N/um^2 modul elastičnosti
nil=0.3;
```

#### Rezultati

```
data1.N = data1.eta * data1.An/1e12;% total nuber of tops per nominal area
data2.N = data2.eta * data2.An/1e12;% total nuber of tops per nominal area
%
[F1, A1, p1, pdf1, Eaux1, Are1, d1]=GW_Model(data1.R, data1.sigma, data1.N,
E1, nil, 10, 1);
[F2, A2, p2, pdf2, Eaux2, Are2, d2]=GW_Model(data2.R, data2.sigma, data2.N,
E1, nil, 10, 1);
```

#### Grafi

```
fprintf('Površina za d=0 pri Ra=0,6 um znaša %.1f um^2.\n',Are1)
fprintf('Površina za d=0 pri Ra=0,06 um znaša %.1f um^2.\n',Are2)

fprintf('Razmerje realne in kontaktne površine pri Ra=0,6 um znaša %.2f %%.\n'
,Are1/data1.An*100)
fprintf('Razmerje realne in kontaktne površine pri Ra=0,06 um znaša %.2f %%.\n'
,Are2/data2.An*100)

figure
hold on
```

```

plot(F1,d1)
plot(F2,d2)
title('Realna verjetna sila na kontaktu')
xlabel('Sila [N]')
ylabel('Položaj referenčne linije glede na srednjo linijo [ $\mu\{m\}$ ]')
legend('R_a = 0,6 \mu\{m}\'', 'R_a = 0,06 \mu\{m}\'')
ca=gca;
ca.YLim=[0,2];
cf=gcf;
cf.Units='normalized';
cf.Position=[0 0 1 1];
grid on
grid minor

figure
hold on
plot(pdf1(d1),d1)
plot(pdf2(d2),d2)
title('Gostota porazdelitve verjetnosti površine')
xlabel('Gostota porazdelitve verjetnosti površine [ $1/\mu\{m\}$ ]')
ylabel('Položaj referenčne linije glede na srednjo linijo [ $\mu\{m\}$ ]')
legend('R_a = 0,6 \mu\{m}\'', 'R_a = 0,06 \mu\{m}\'')
ca=gca;
ca.YLim=[0,2];
cf=gcf;
cf.Units='normalized';
cf.Position=[0 0 1 1];
grid on
grid minor

figure
hold on
plot(A1,d1)
plot(A2,d2)
title('Realna verjetna površina v stiku')
xlabel('Površina v stiku [ $\mu\{m\}^2$ ]')
ylabel('Položaj referenčne linije glede na srednjo linijo [ $\mu\{m\}$ ]')
legend('R_a = 0,6 \mu\{m}\'', 'R_a = 0,06 \mu\{m}\'')
ca=gca;
ca.YLim=[0,2];
cf=gcf;
cf.Units='normalized';
cf.Position=[0 0 1 1];
grid on
grid minor

figure
hold on
plot(pl.*1e6,d1)% faktor je zaradi N/um^2 \to MPa

```

```
plot(p2.*1e6,d2)% faktor je zaradi N/um^2 \to MPa
title('Stični tlak v kontaktu')
xlabel('Stični tlak v kontaktu [MPa]')
ylabel('Oddaljenost od srednje linije d [\mu{m}]')
legend('R_a = 0,6 \mu{m}', 'R_a = 0,06 \mu{m}')
cf=gcf;
cf.Units='normalized';
cf.Position=[0 0 1 1];
grid on
grid minor

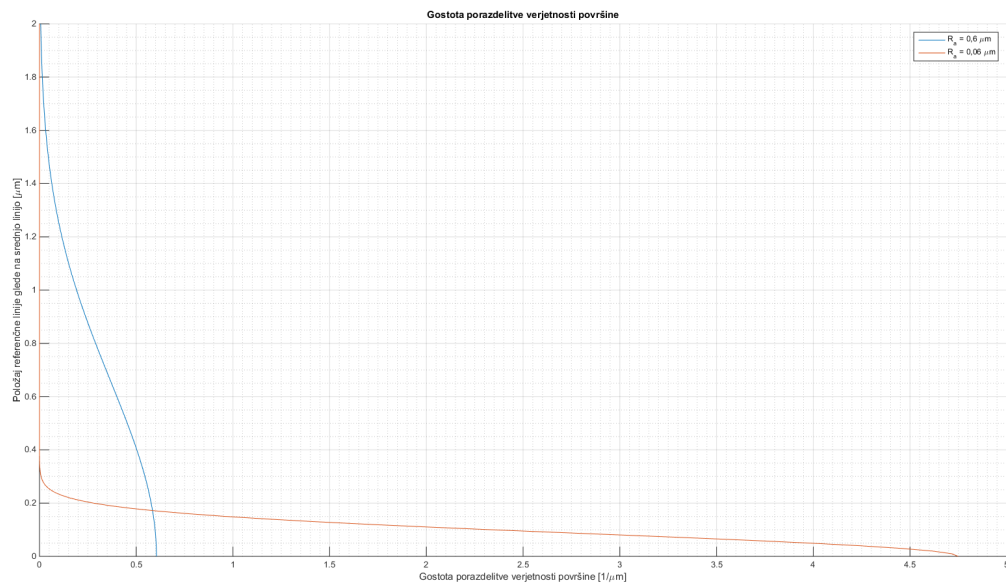
figure
hold on
plot(F1,A1./data1.An)
plot(F2,A2./data2.An)
title('Razmerje površin za glede na silo')
xlabel('Kontaktna sila F_c [N]')
ylabel('Razmerje realne z nominalno kontaktno površino A_i/A_{re}')
legend('R_a = 0,6 \mu{m}', 'R_a = 0,06 \mu{m}', 'Location', 'southeast')
cf=gcf;
cf.Units='normalized';
cf.Position=[0 0 1 1];
grid on
grid minor
```

Površina za  $d=0$  pri  $R_a=0,6 \mu\text{m}$  znaša  $3325.7 \mu\text{m}^2$ .

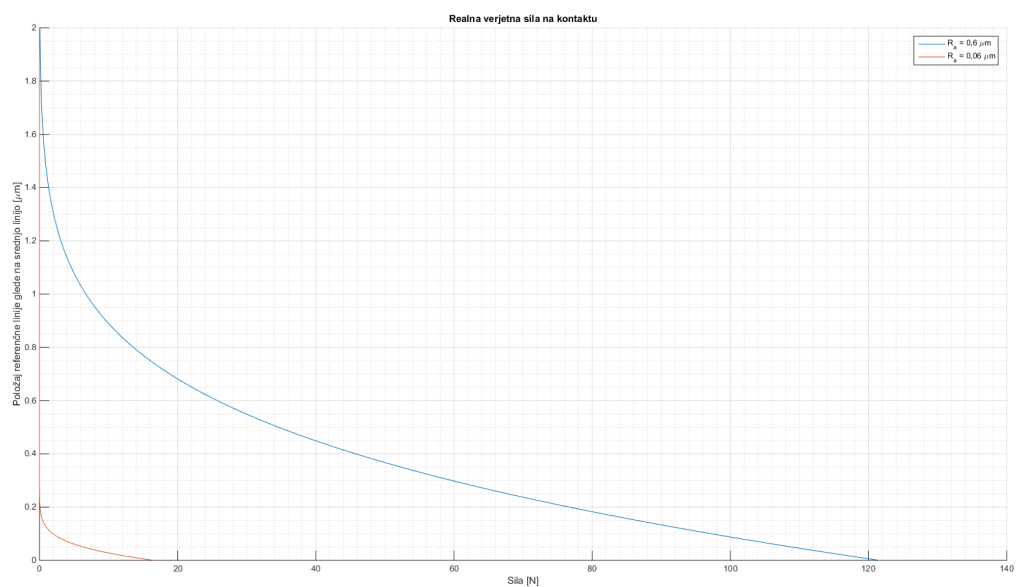
Površina za  $d=0$  pri  $R_a=0,06 \mu\text{m}$  znaša  $2231.8 \mu\text{m}^2$ .

Razmerje realne in kontaktne površine pri  $R_a=0,6 \mu\text{m}$  znaša  $7.73 \%$ .

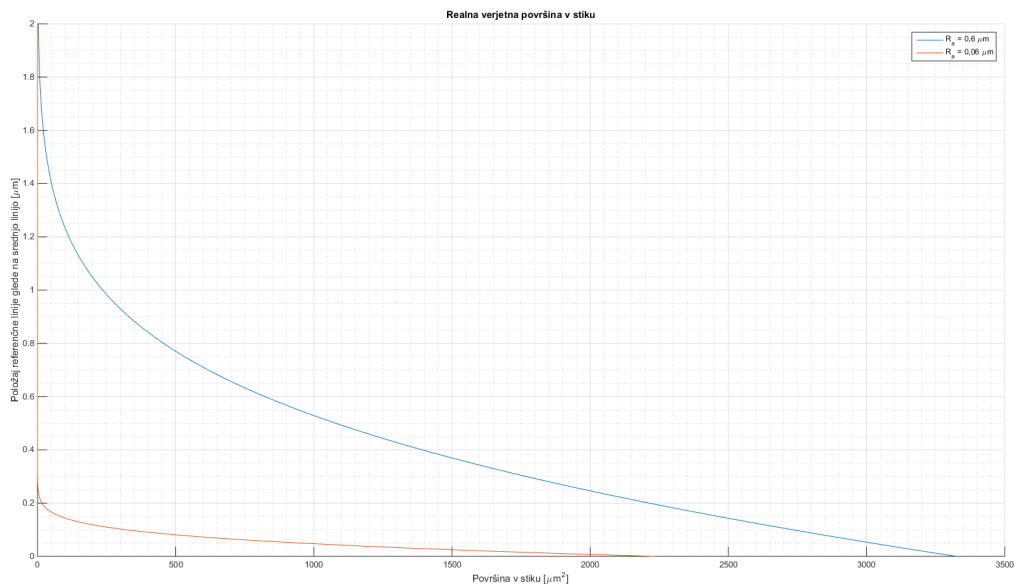
Razmerje realne in kontaktne površine pri  $R_a=0,06 \mu\text{m}$  znaša  $5.19 \%$ .



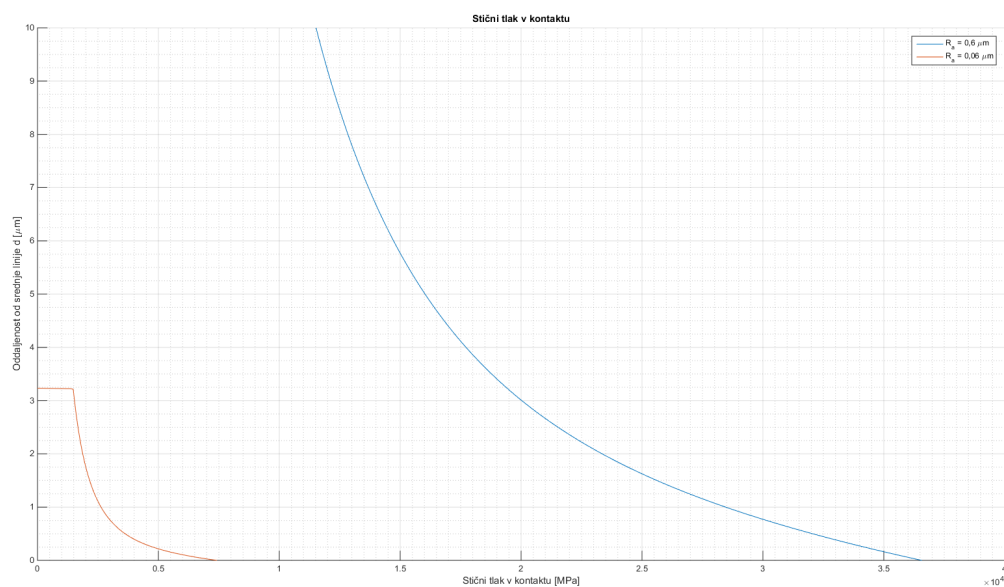
Slika 1: Prikaz funkcije gostote porazdelitve hrapavostnega profila



Slika 2: Prikaz verjetne kontaktne sile

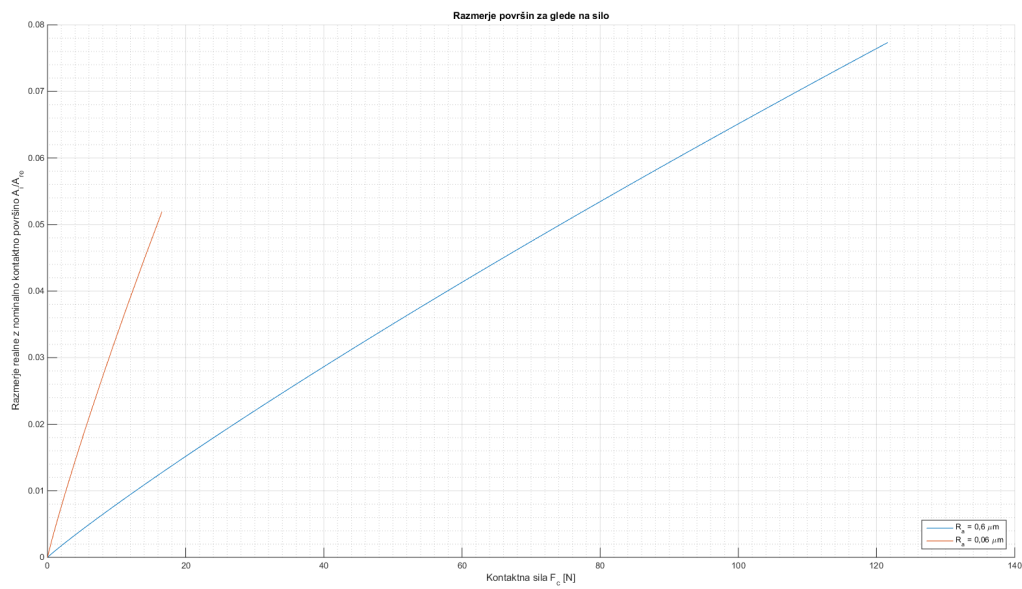


Slika 3: Prikaz verjetne kontaktne površine



Slika 4: Prikaz verjetnega povprečnega stičnega tlaka





Slika 5: Prikaz razmerja realne površine glede na nominalno površino glede na aplicirano silo

### 1.2.2 Metoda 2 - glavni program in rezultati

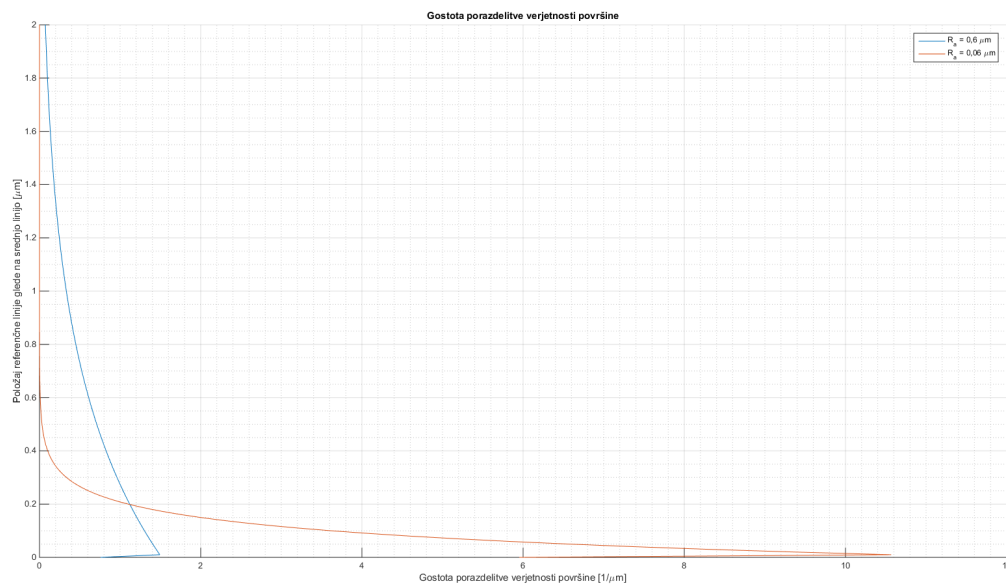
Razlika v programu med metodo 1 - normalno porazdelitvijo gostote verjetnosti izmerkov glede na metodo 2 - eksponentno porazdelitvijo gostote verjetnosti izmerkov je samo v eni številki - ko program kliče funkcijo *GW\_Model* je zadnji parameter številka 2 namesto številke 1, to spremeni funkcije porazdelitve gostote verjetnosti v tej funkciji, zato ponoven program ne bo ponovljen; prikazani so samo rezultati.

Površina za  $d=0$  pri  $R_a=0,6 \mu\text{m}$  znaša  $8336.3 \mu\text{m}^2$ .

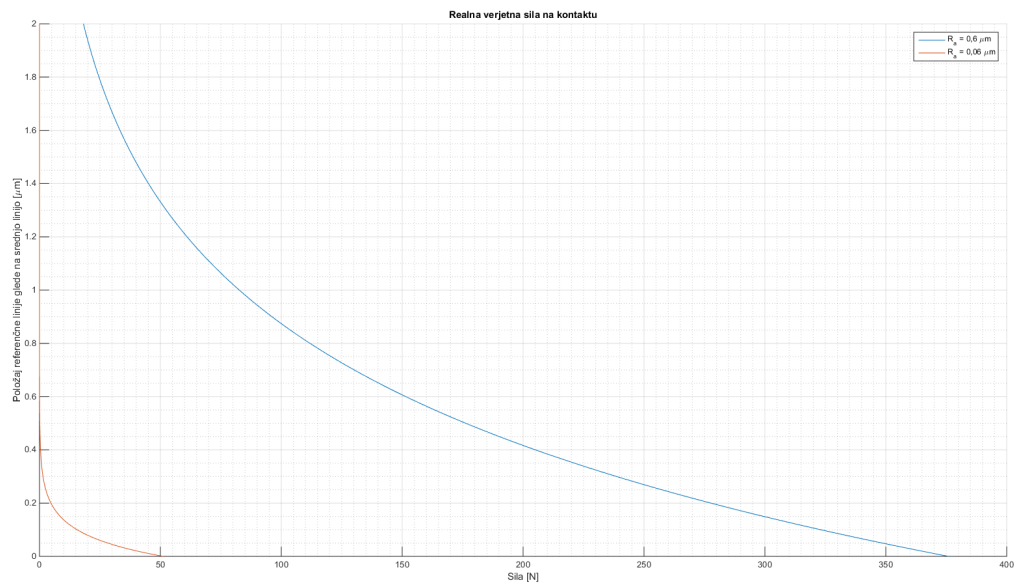
Površina za  $d=0$  pri  $R_a=0,06 \mu\text{m}$  znaša  $5594.3 \mu\text{m}^2$ .

Razmerje realne in kontaktne površine pri  $R_a=0,6 \mu\text{m}$  znaša 19.39 %.

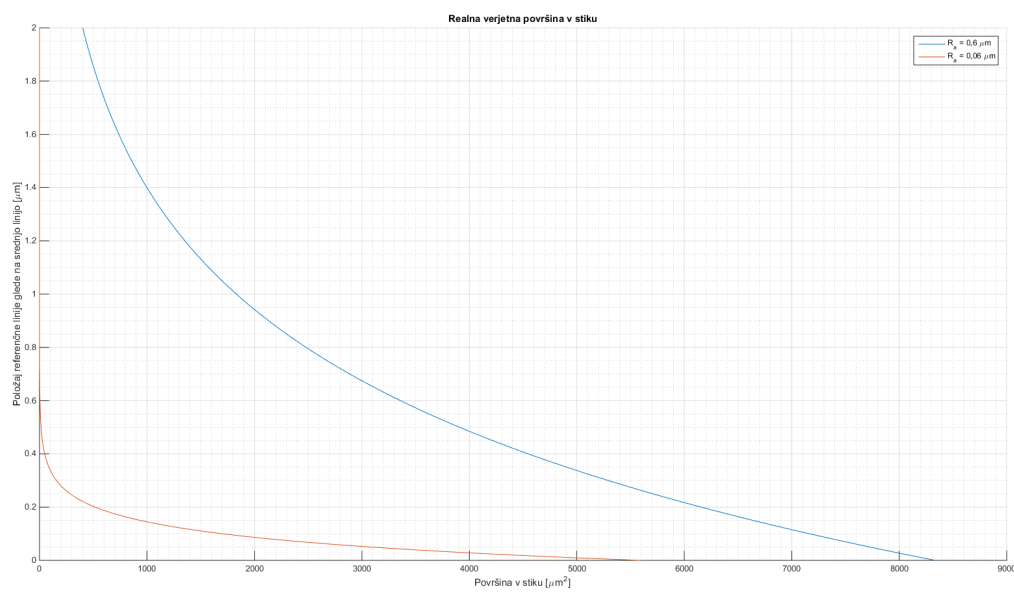
Razmerje realne in kontaktne površine pri  $R_a=0,06 \mu\text{m}$  znaša 13.01 %.



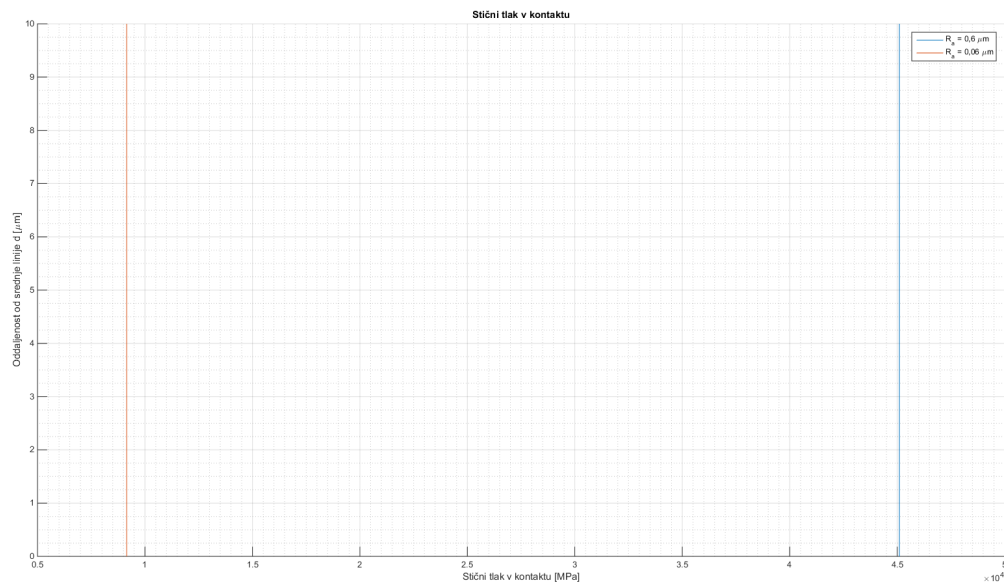
Slika 6: Prikaz funkcije gostote porazdelitve hrapavostnega profila



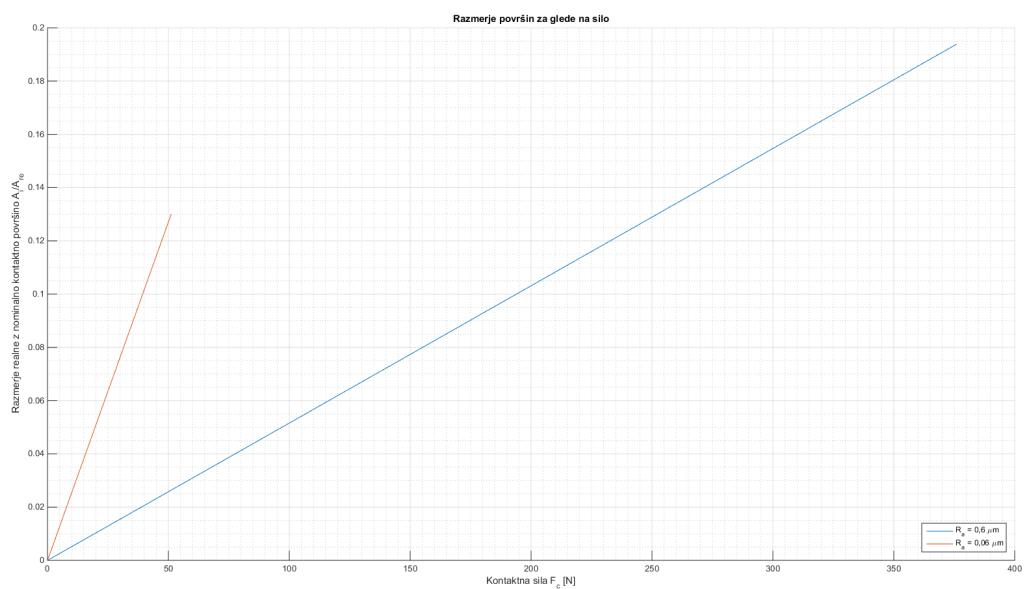
Slika 7: Prikaz verjetne kontaktne sile



Slika 8: Prikaz verjetne kontaktne površine



Slika 9: Prikaz verjetnega povprečnega stičnega tlaka



Slika 10: Prikaz razmerja realne površine glede na nominalno površino glede na aplicirano silo

### 1.2.3 Funkcija GW\_model

```
function [F, A, p, pdf, Eaux, Are, d]=GW_Model(R, sigma, N, E1, nil, konec,
method)
```

#### Glavni izračun

```
Eaux = 1/((1-nil^2)/(E1)); % nadomestni modul elastičnosti % predpostavljamo,
da je druga površina popolnoma toga
```

#### Vhodne enačbe

```
eqnFbeg =@(z,d) (4/3) * Eaux * sqrt(R) * N * (z-d).^ (3/2); %force
probability equation begining

eqnAbeg =@(z,d) N * pi * R * (z-d); % area probability equation begining
switch method
case 1 % normal distribution
mu=0;%mean distribution value
pdf=@(z) (1 / (sqrt(2 * pi * sigma^2))) .* exp(-(((z-mu).^2)/(2*sigma^2)));
eqnF =@(z,d) eqnFbeg(z,d).* pdf(z); %force probability equation
eqnA =@(z,d) eqnAbeg(z,d).* pdf(z); % area probability equation
case 2 % exponential distribution
pdf=@(z) (1 / sigma)*exp((-1/sigma)*z).*heaviside(z);
eqnF =@(z,d) eqnFbeg(z,d).* pdf(z); %force probability equation
eqnA =@(z,d) eqnAbeg(z,d).* pdf(z); % area probability equation
otherwise
fprintf('Taka porazdelitev gostote verjetnosti ne obstaja.\n')
return
end
```

#### Izračuni

```
idx=1;%indeksirnik
zacetek=0;
delitev=1000;
F=zeros(delitev,1);
A=zeros(delitev,1);
p=zeros(delitev,1);
for d = linspace(zacetek,konec,delitev)
F(idx)= integral(@(z)eqnF(z,d),d,inf);% sila ki se verjetno pojavi v kontaktu
A(idx)= integral(@(z)eqnA(z,d),d,inf);% verjetna površina kontakta
p(idx)= F(idx)/A(idx);% verjetni tlak v kontaktu
idx = idx+1;% indeksirnik za zapis rezultatov v vektor
end
Are=A(1);% Ob predpostavki, da je zacetek==0
d=linspace(zacetek,konec,delitev);% zato da mi lahko izpiše delitev kot output
end
```

## 1.3 Zbirnik rezultatov

Rezultati so zaradi večje preglednosti zbrani v tabelah 1 in 2.

Nominalna hrapavost	$R_a = 0,6\mu\text{m}$	$R_a = 0,06\mu\text{m}$
Kontaktna površina za $d = 0$ [ $\mu\text{m}$ ]	3325.7	2231.8
Razmerje realne in kontaktne površine [%]	7.73	5.19

Tabela 1: Glavni izračunani parametri za normalno porazdelitev gostote verjetnosti profila hrapavosti

Nominalna hrapavost	$R_a = 0,6\mu\text{m}$	$R_a = 0,06\mu\text{m}$
Kontaktna površina za $d = 0$ [ $\mu\text{m}$ ]	8336.3	5594.3
Razmerje realne in kontaktne površine [%]	19.39	13.01

Tabela 2: Glavni izračunani parametri za eksponentno porazdelitev gostote verjetnosti profila hrapavosti

## 1.4 Zaključek - realna kontaktna površina

Zaradi manjše numerične napake je smiselno vse vhodne podatke transformirati tako, da so nastopajoči podatki bližje številki 1. S tem se izognemo težavi zapisovanja operacij med zelo velikimi in zelo majhnimi števili - numerični pristop ima končno natančnost. Kljub temu pa je zaradi dejstva, da normalna porazdelitev gostote verjetnosti konvergira k vrednosti 0 naletimo na težave pri računanju, ki so povezane z numeričnim pristopom reševanja<sup>1</sup>. To dejstvo je dobro vidno na sliki 4, kjer je možno opaziti nezvezen zlom grafa.

Grafa gostote porazdelitve verjetnosti hrapavostnega profila nista presenetljiva - večja kot je hrapavost, bolj je krivulja "široka" oz. njene vrednosti so porazdeljene po večjem intervalu neodvisne spremenljivke - položaja.

Grafa sil prikazujeta da se sila začne povečevati prej pri večji oddaljenosti ploskve od referenčne razdalje pri večji hrapavosti, kar je logično, saj je verjetnost da prej zadenemo del površine pri večji hrapavosti večja. Ko dosežemo s površino srednjo linijo je sila pri večji hrapavosti večja od manjše hrapavosti, saj je deformirane več površine, gledano lateralno. Podobno velja tudi za graf površine v odvisnosti od višine do katere poteka deformacija. Pri eksponentni porazdelitvi je sila večja zaradi narave porazdelitve - vsa površina je nad srednjo linijo.

Stični tlak pada z oddaljenostjo od srednje linije pri Gaussovi porazdelitvi, vizualno to pomeni, da bolj kot obremenjujemo površini, vse večje napetosti apliciramo nanje. Nasprotno pri eksponentni porazdelitvi pa je tlak izračunan kot konstanten po celotnem območju, to so utemeljili Johnson, Greenwood in Poon [1] da je tlak praktično neodvisen od aplicirane sile. Dejstvo za tem je, da se s povečevanjem obremenitve povečuje kontaktna površina. Težave v tem odseku so samo s predpostavko, da se vršički deformirajo elastično, logično je, da ko presežemo neko mejo, kjer je deformacija pretežno plastična tlak ni več konstanten, v mejnem primeru, ko bi realna kontaktna površina bila enaka nominalni pa bi tlak naraščal linearno z obremenitvijo, kot to predvideva model  $p = F/A$ . Opaziti je možno tudi da so tlaki pri vrednosti  $d=0$  bistveno večji pri eksponentni kot pri normalni porazdelitvi, kar lahko implicira v predimenzionirane nosilne elemente.

Razmerja realne kontaktne površine z nominalno povedo, da je ob predpostavki eksponentne porazdelitve graf "podaljšek" grafa normalne porazdelitve, namreč vrednosti se približno ujemajo pri obeh porazdelitvah, seveda z izjemo, da očitno eksponentna porazdelitev nosi bistveno več kot normalna - konec grafa predstavlja točko  $d = 0$ . Postavitev grafov je logična - manjša hrapavost pomeni, da se mora deformirati manj površine, da bi dosegli enako razmerje realne in nominalne stične površine, posledično mora biti aplicirana sila pri manjši hrapavosti manjša.

<sup>1</sup>Najmanjše možno pozitivno število v načinu plavajoče vejice v dvojni natančnosti je približno  $2.2251 \cdot 10^{-308}$

## 2. Realna kontaktna temperatura

### 2.1 Uvod

Realna kontaktna temperatura je pojem, ki se nanaša na povprečno notranjo energijo gradnikov materiala, ki so neposredno v stiku z drugim telesom. Ker je v splošnem narava površine kaotična, je praktično neobhodno, da se omejimo na določene predpostavke, na katerih gradimo matematični model. Temperatura kontakta dveh teles <sup>2</sup> so posledica dissipacije mehanskega dela preko trenja v toploto [1], kateri lahko sledi lokalna sprememba lastnosti materiala. To hipno spremembo imenujemo “flash” temperatura, ki je posebno izrazita pri suhem drsnem trenju.

Temperaturo kontakta definiramo kot:

$$T_c = T_0 + T_{fl} \quad (5)$$

Kjer  $T_0$  označuje začetno temperaturo mase pred drsenjem,  $T_c$  temperaturo kontakta po zdrsu in  $T_{fl}$  flash temperaturo.

Predpostavke na katerih je zgrajen model [1]:

- Kontaktna točka je obravnavana kot ravna ploskev,
- torna toplota se generira enakomerno po celotni ploskvi,
- toplota se izmenjuje samo skozi proces prevoda skozi trdnine,
- torni koeficient je konstanten,
- telesi sta pred tem pojavom v termičnem ravnovesju.

Zavestno je bilo zanemarjena predpostavka, da naj bodo snovne lastnosti tekom procesa konstantne. Privzeto je bilo, da se prevodnostni koeficient  $\lambda$  spreminja s temperaturo. To zahteva numerično reševanje postopka, pri katerem je bila uporabljena navadna iteracija.

Izkaže se, da je temperatura bistveno odvisna od gradienta temperaturnega polja po materialu. Za ta namen se vpelje brezdimenzijsko Pecletovo število, ki ponazarja gradient [1]. Če je dovolj majhno je gradient majhen in obratno. To dejstvo je bilo zajeto v izračunu v notranji *if* zanki.

### 2.2 Glavni program

Zaradi zanimivosti reševanja je bil postopek deljen na dva, medsebojno ločena dela oz. metodi. Prva metoda podatkom priredi kvadratno parabelo - vse vmesne vrednosti so torej interpolirane glede na te tri znane vrednosti koeficienta prevodnosti. Druga metoda pa podatkom priredi odsekovno linearno funkcijo, krajše, iz podatkov naredi linearne zlepkke. Vse te funkcije so zbrane v Matlabovi funkciji *fit*.

Medsebojna primerjava prve in druge metode da sklep, da sta metodi skoraj ekvivalentni, saj lahko ocenimo, da so podatki razporejeni približno linearno (koeficient pri kvadratnem členu parabole je zelo blizu 0). To potrjujejo tudi izračunani parametri. Rezultati so predstavljeni v tabeli 3. Sam postopek - funkcijski diagram pa je predstavljen na sliki 12.

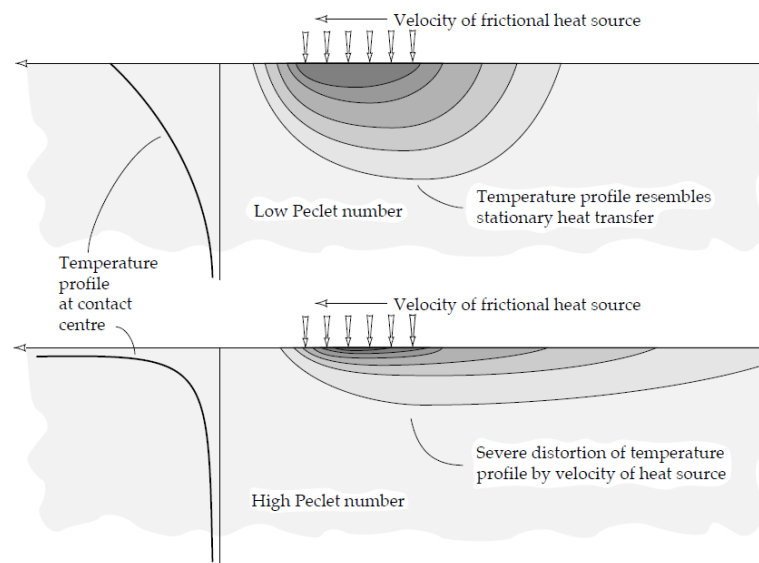
Rezultati, podani v tabeli 3 povedo, da je temperatura dosežena pri višji hitrosti višja, kar je tudi pričakovano, saj je toplotna moč, povezana preko sile trenja višja, prav tako pa so gradienti višji, kar tudi predvideva Pecletovo število.

#### 2.2.1 Metoda 1

```
clc
clear
close all
```

---

<sup>2</sup>ang. *surface conjunction temperature*



Slika 11: Ponazoritev Pecletovega števila

### Vhodni podatki

```

F = 15; %[N] normal force
mu = 0.1; %[1] coefficient of friction
d = 12.7e-3; %[m] ball diameter
rho = 7850; %[kg/m^3] material density
c = 460; %[J/kgK] heat capacity
v1 = 0; %[m/s] first body velocity
v2(1) = 0.1; %[m/s] second body velocity 1
v2(2) = 0.8; %[m/s] second body velocity 2
E1 = 210e9; %[Pa] Youngs modulus
ni1 = 0.3; %Poisson number
E2 = 210e9; %[Pa] Youngs modulus
ni2 = 0.3; %Poisson number
T_Ref = [0, 400, 600]; %[°C] Reference temperature for heat flux coefficients
lamRef = [59, 44, 37]; %[W/mK] heat flux coefficient
T0 = 0; %[°C] room temperature
initT = 1; %[°C] begining temperature in iteration process
limVal=200; % max. iterations

```

### Mehanski in kinematični izračuni

```

del_v = abs(v2-v1); %Relative velocity vector
R = d/2; %ball radius
Eaux = ((1-ni1^2)/(E1) + (1-ni1^2)/(E1))^-1; % Auxillary Youngs modulus
a = ((3*F*R)/(4*Eaux))^(1/3); % Contact radius - Hertz theory

```

### Interpolacija koeficienta prevodnosti

```

method = 1; %1 - 2nd degree polynomial, 2-linear by intervals
switch method

```



```

case 1 % data is fitted with a 2nd degree polynomial
[xData, yData] = prepareCurveData( T_Ref, lamRef );
ft = fitttype( 'poly2' );
[fitresult, gof] = fit( xData, yData, ft );
fprintf('Parametri interpoliranja so:\n')
disp(gof)
% Plot fit with data.
figure( 'Name', 'myFit' );
h = plot( fitresult, xData, yData );
legend( h, '\lambda vs. T_{Ref}', 'Interpolacija', 'Location', 'NorthEast' );
xlabel('Referenčna temperatura [^\circ{C}]')
ylabel('Interpoliran koeficient prevodnosti \lambda [W/(mK)]')
grid on
grid minor
ca=gcf;
ca.Units='normalized';
ca.Position = [0 0 1 1];
case 2 % data is fitted with section-linear fit
[xData, yData] = prepareCurveData( T_Ref, lamRef );
ft = 'linearinterp';
[fitresult, gof] = fit( xData, yData, ft, 'Normalize', 'on' );
fprintf('Parametri interpoliranja so:\n')
disp(gof)
figure( 'Name', 'myFit' );
h = plot( fitresult, xData, yData );
legend( h, '\lambda vs. T_{Ref}', 'Interpolacija', 'Location', 'NorthEast' );
xlabel('Referenčna temperatura [^\circ{C}]')
ylabel('Interpoliran koeficient prevodnosti \lambda [W/(mK)]')
grid on
grid minor
ca=gcf;
ca.Units='normalized';
ca.Position = [0,0,1,1];
end

```

Parametri interpoliranja so:

```

    sse: 1.0097e-28
  rsquare: 1
    dfe: 0
adjrsquare: NaN
    rmse: NaN

```

### Zanka za izračun

```

figure
for idx1=1:numel(del_v)% In each del_v at once
    v=del_v(idx1);% v je hitrost v while zanki
    Tf1=initT;% initial values for while loop to function

```

```

Told=initT+1;
idx=1;
TT(1)=initT;
while abs(Told-Tfl)>0.001 % absolute error between iterations
    Told=Tfl;
    lam=fitresult(Told);% flux coefficient in this iteration
    L=(v*a*rho*c)/(2*lam);%Peclet number for this iteration
    if L < 0.1 % zelo počasno širjenje toplote
        Tfl = (F * mu * v) / (4 * a * lam);
    elseif L > 5 % zelo hitro širjenje toplote
        Tfl = (F * mu * v)/(3.25 * a * lam) * sqrt((1)/(2*L));
    else % nekaj vmes - predpostavimo da interpoliramo vrednosti
        Tfl_sp = (F * mu * v) / (4 * a * lam);%L<0.1
        Tfl_zg = (F * mu * v)/(3.25 * a * lam) * sqrt((1)/(2*L));%L>5
        Tfl=interp1([0.1,5],[Tfl_sp, Tfl_zg],L,'linear');%vmesna vrednost
    end
    % za izris konvergence in blokado, če je več kot 400 korakov
    TT(idx+1)=Tfl;
    idx=idx+1;
    if idx>limVal
        fprintf('Exceeded max. iterations. Going in next one ...')
        TT(idx)=NaN;
        break
    end
end
LL(idx1)=L;% Peclet number for final results
Tfl_v(idx1)=Tfl;% Flash temperature for final results
% Plotting the iterations
hold on
plot(TT,'.')
xlabel('Številka iteracije')
ylabel('Flash temperatura [°C]')
grid on

end
leg=['Flash temperatura za v=',num2str(del_v(1)), ' m/s';
'Flash temperatura za v=',num2str(del_v(2)), ' m/s'];
legend(leg,'Location','northeastoutside')
ca=gcf;
ca.Units='normalized';
ca.Position = [0,0,1,1];

```

### Kontaktna temperatura

```

Tc=Tfl_v+T0;% contact temperature vector
fprintf('Kontaktna temperatura je %.2f °C.\n',Tc)
fprintf('Temperatura okolice je %.1f °C.\n',T0)

```

Kontaktna temperatura je 32.67 °C.

Kontaktna temperatura je 76,16 °C.  
 Temperatura okolice je 25.0 °C.

### 2.2.2 Metoda 2

#### Vhodni podatki

```
F = 15; %[N] normal force
mu = 0.1; %[1] coefficient of friction
d = 12.7e-3; %[m] ball diameter
rho = 7850; %[kg/m^3] material density
c = 460; %[J/kgK] heat capacity
v1 = 0; %[m/s] first body velocity
v2(1) = 0.1; %[m/s] second body velocity 1
v2(2) = 0.8; %[m/s] second body velocity 2
E1 = 210e9; %[Pa] Youngs modulus
ni1 = 0.3; %Poisson number
E2 = 210e9; %[Pa] Youngs modulus
ni2 = 0.3; %Poisson number
T_Ref = [0, 400, 600]; %[°C] Reference temperature for heat flux coefficients
lamRef = [59, 44, 37]; %[W/mK] heat flux coefficient
T0 = 0; %[°C] room temperature
initT = 1; %[°C] begining temperature in iteration process
limVal=200; % max. iterations
```

#### Mehanski in kinematični izračuni

```
del_v = abs(v2-v1); %Relative velocity vector
R = d/2; %ball radius
Eaux = Eaux = ((1-ni1^2)/(E1) + (1-ni1^2)/(E1))^-1; % Auxillary Youngs modulus
a = ((3*F*R)/(4*Eaux))^(1/3); % Contact radius - Hertz theory
```

#### Interpolacija koeficienta prevodnosti

```
method = 2; %1 - 2nd degree polynomial, 2-linear by intervals
switch method
case 1 % data is fitted with a 2nd degree polynomial
[xData, yData] = prepareCurveData( T_Ref, lamRef );
ft = fitype( 'poly2' );
[fitresult, gof] = fit( xData, yData, ft );
fprintf('Parametri interpoliranja so:\n')
disp(gof)
% Plot fit with data.
figure( 'Name', 'myFit' );
h = plot( fitresult, xData, yData );
legend( h, '\lambda vs. T_{Ref}', 'Interpolacija', 'Location', 'NorthEast' );
xlabel('Referenčna temperatura [^\circ{C}]')
ylabel('Interpoliran koeficient prevodnosti \lambda [W/(mK)]')
grid on
grid minor
```

```

ca=gcf;
ca.Units='normalized';
ca.Position = [0 0 1 1];
case 2 % data is fitted with section-linear fit
[xData, yData] = prepareCurveData( T_Ref, lamRef );
ft = 'linearinterp';
[fitresult, gof] = fit( xData, yData, ft, 'Normalize', 'on' );
fprintf('Parametri interpoliranja so:\n')
disp(gof)
figure( 'Name', 'myFit' );
h = plot( fitresult, xData, yData );
legend( h, '\lambda vs. T_{Ref}', 'Interpolacija', 'Location', 'NorthEast' );
xlabel('Referenčna temperatura [^\circ{C}]')
ylabel('Interpoliran koeficient prevodnosti \lambda [W/(mK)]')
grid on
grid minor
ca=gcf;
ca.Units='normalized';
ca.Position = [0,0,1,1];
end

```

Parametri interpoliranja so:

```

sse: 0
rsquare: 1
dfe: 0
adjrsquare: NaN
rmse: NaN

```

### Zanka za izračun

```

figure
for idx1=1:numel(del_v)% In each del_v at once
v=del_v(idx1);% v je hitrost v while zanki
Tfl=initT;% initial values for while loop to function
Told=initT+1;
idx=1;
TT(1)=initT;
while abs(Told-Tfl)>0.001 % absolute error between iterations
Told=Tfl;
lam=fitresult(Told);% flux coefficient in this iteration
L=(v*a*rho*c)/(2*lam);%Peclet number for this iteration
if L < 0.1 % zelo počasno širjenje toplote
Tfl = (F * mu * v) / (4 * a * lam);
elseif L > 5 % zelo hitro širjenje toplote
Tfl = (F * mu * v)/(3.25 * a * lam) * sqrt((1)/(2*L));
else % nekaj vmes - predpostavimo da interpoliramo vrednosti
Tfl_sp = (F * mu * v) / (4 * a * lam);%L<0.1
Tfl_zg = (F * mu * v)/(3.25 * a * lam) * sqrt((1)/(2*L));%L>5

```

```

Tfl=interp1([0.1,5],[Tfl_sp, Tfl_zg],L,'linear');%vmesna vrednost
end
% za izris konvergence in blokado, če je več kot 400 korakov
TT(idx+1)=Tfl;
idx=idx+1;
if idx>limVal
fprintf('Exceeded max. iterations. Going in next one ...')
TT(idx)=NaN;
break
end
end
LL(idx1)=L;% Peclet number for final results
Tfl_v(idx1)=Tfl;% Flash temperature for final results
% Plotting the iterations
hold on
plot(TT,'.')
xlabel('Številka iteracije')
ylabel('Flash temperatura [°C]')
grid on
end
leg=['Flash temperatura za v=',num2str(del_v(1)),
' m/s'; 'Flash temperatura za v=',num2str(del_v(2)), ' m/s'];
legend(leg,'Location','northeastoutside')
ca=gcf;
ca.Units='normalized';
ca.Position = [0,0,1,1];

```

### Kontaktna temperatura

```

Tc=Tfl_v+T0;% contact temperature vector
fprintf('Kontaktna temperatura je %.2f °C.\n',Tc)
fprintf('Temperatura okolice je %.1f.\n',T0)

```

Kontaktna temperatura je 32.67 °C.  
Kontaktna temperatura je 76,12 °C.  
Temperatura okolice je 25.0 °C.

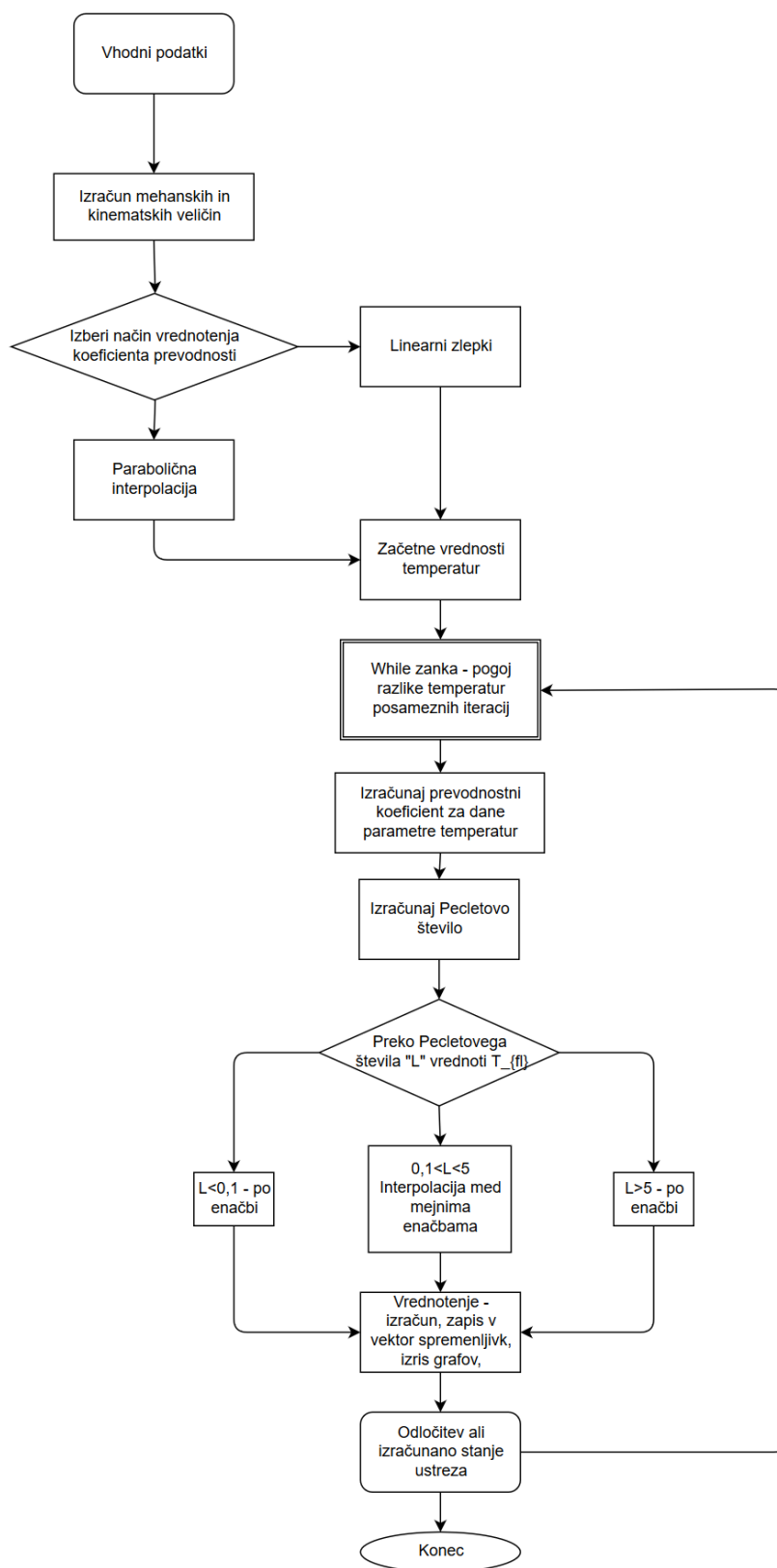
	Metoda 1 - parabola	Metoda 2 - linearni zleпки
$T_c(v = 0,1 \text{ m/s})[°C]$	32,67	32,67
$T_c(v = 0,8 \text{ m/s})[°C]$	76,16	76,12

Tabela 3: Rezultati izmerkov

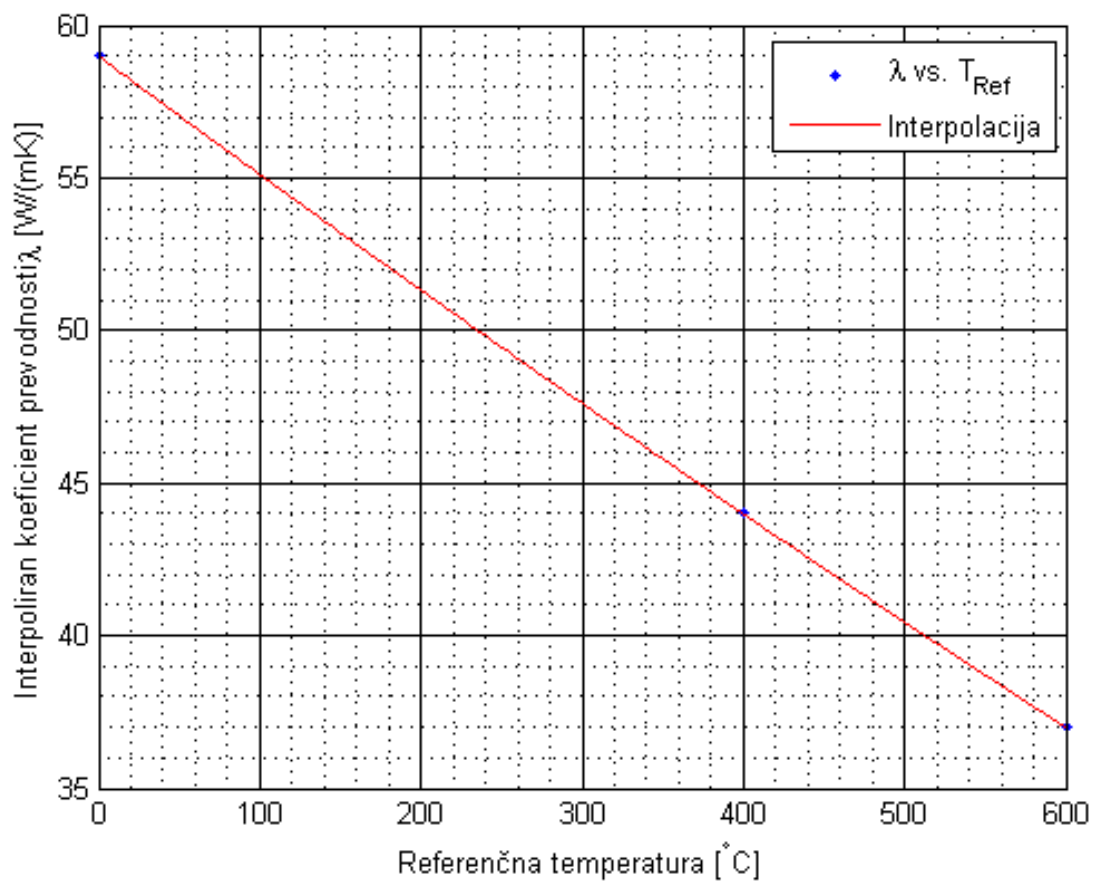
### 2.3 Zaključek - temperature v kontaktu

Glede na izračunane temperature in kombinacijo materialov je možno sklepati, da enkratna ponovitev dogodka drsenja ne povzroči bistvenih sprememb v mikrostrukturi zaradi vpliva temperature. Ob ponavljajočih dogodkih pa bi lahko nadaljnje povečevanje vodilo do sprememb. Nenavadno pri tovrstnem izračunu je, da

se iteracija vrši samo po flash temperaturi, ne pa tudi hkrati z okoliško temperaturo. Zaslediti je mogoče, da se viri [1] omejujejo na konstantne materialne lastnosti tekom dogodka, kar anulira poskus rešitve, ki jo navaja ta dokument.

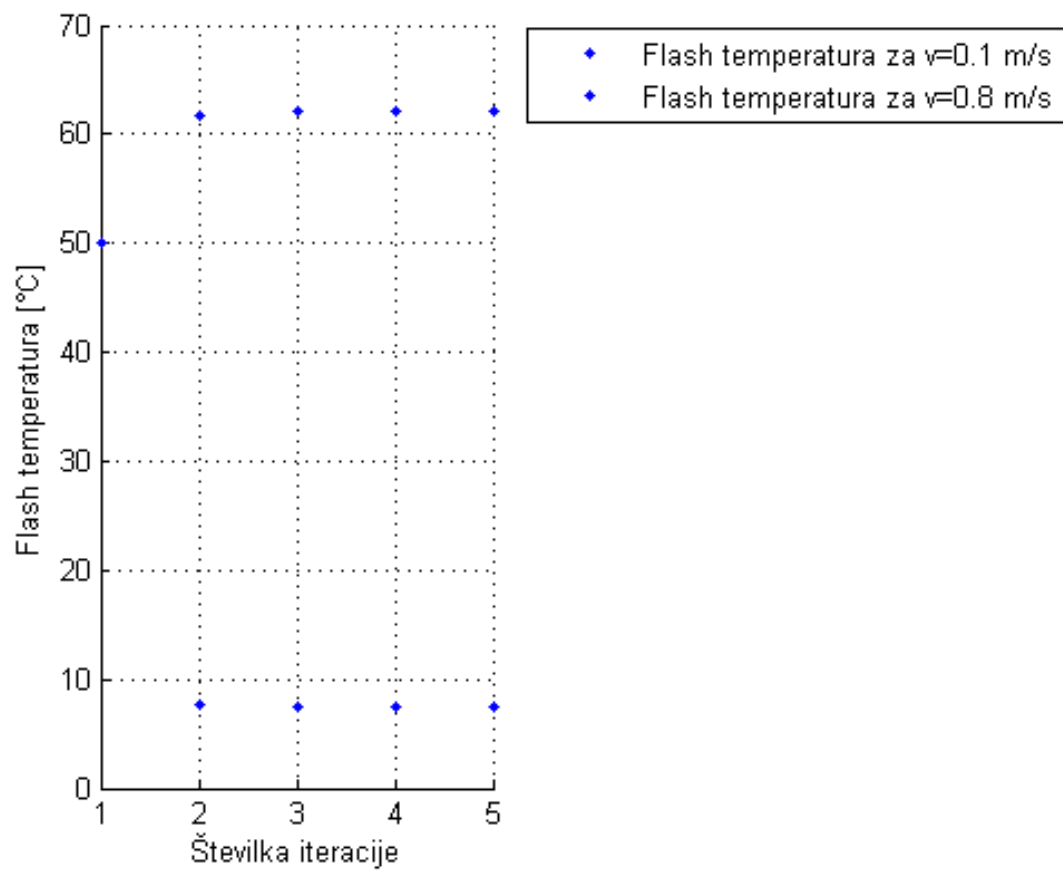


Slika 12: Diagram poteka programa [1]

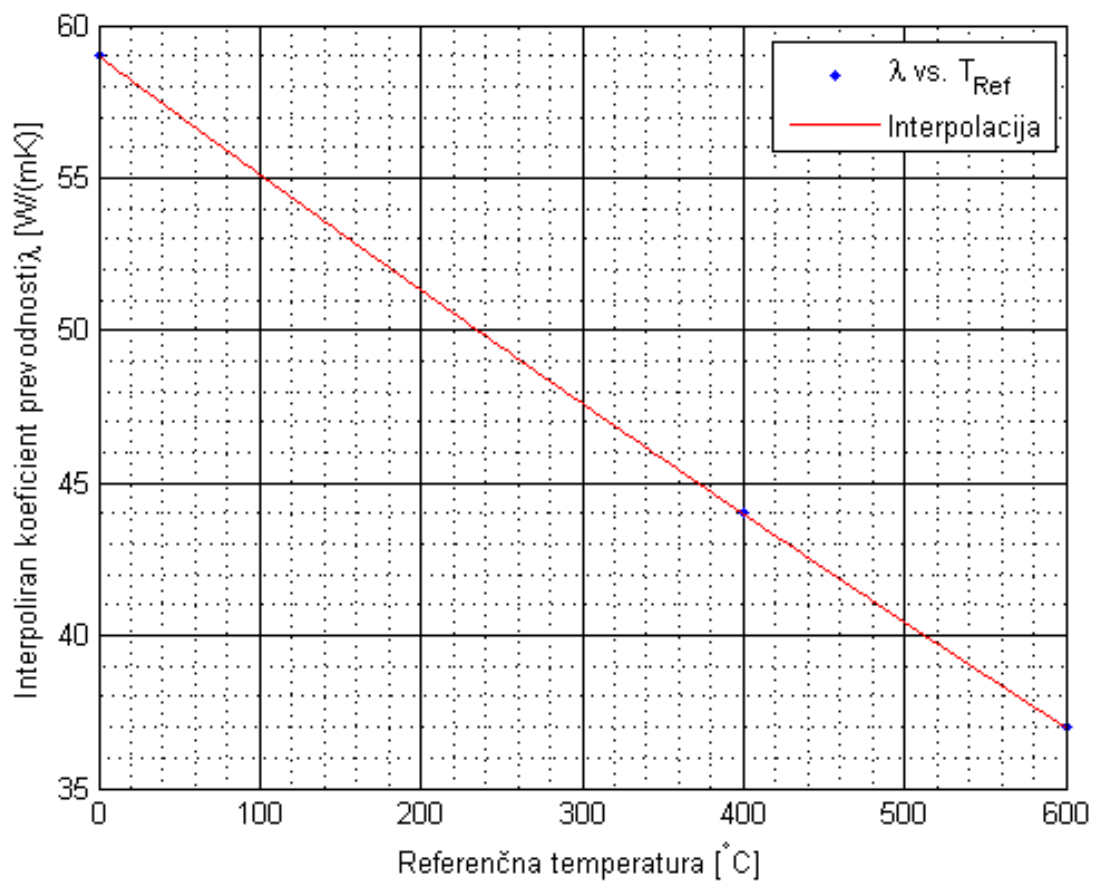


Slika 13: Graf prikaza interpolacije - metoda 1

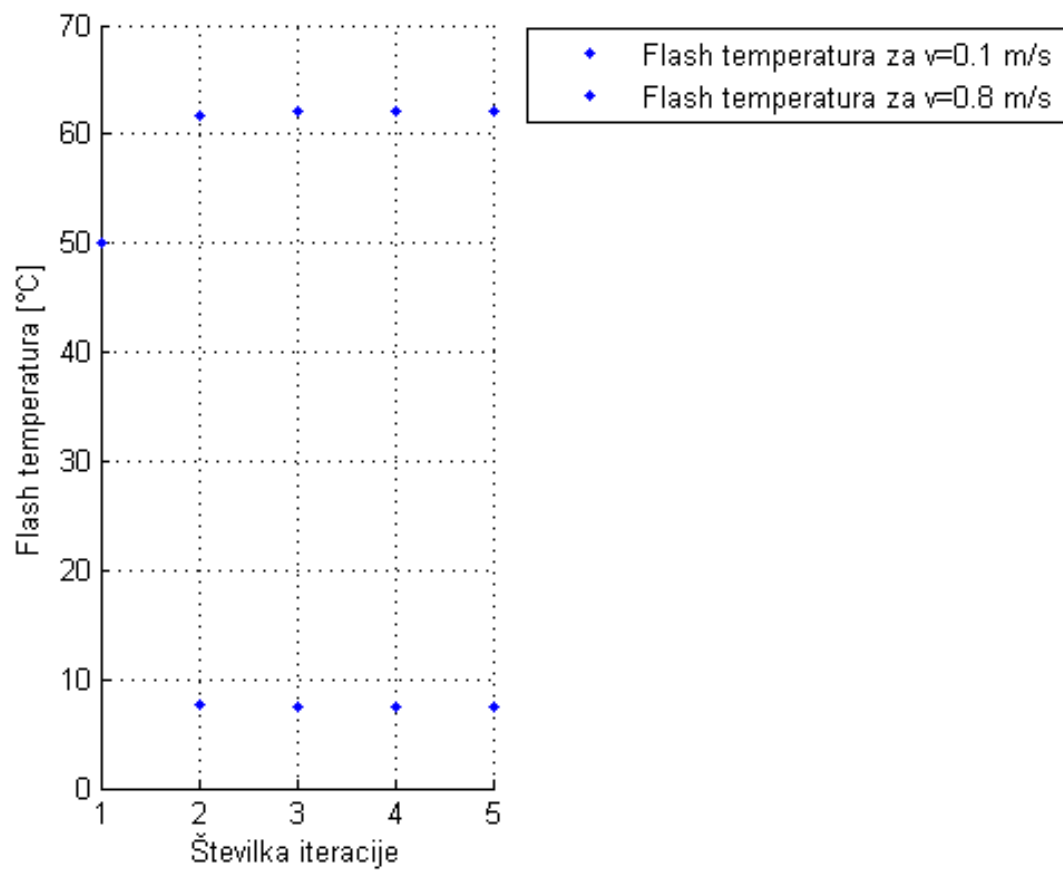




Slika 14: Prikaz konvergiranj vrednosti Flash temperature - metoda 1



Slika 15: Graf prikaza interpolacije - metoda 2



Slika 16: Prikaz konvergiranj vrednosti Flash temperature - metoda 2

## Literatura

- [1] A. W. B. Gwidon W. Stachowiak: *Engineering tribology*. Butterworth Heinemann, 2013.
- [2] TINT: *Diapozitivi za vaje pri predmetu tribologija*.
- [3] G. M. H. M. I. K. Bronštejn, K. A. Semendjajev: *Matematični priročnik*. Ljubljana: TZS, 2012.
- [4] H. R. R. Gohar: *Fundamentals of tribology*. Imperial College Press, 2008.