Ivana Medoš i Jelena Pejković

Širina distribucije intenziteta svetlosti odbijene od neravne metalne površi kao mera njene hrapavosti

Pri odbijanju svetlosti od hrapave površi metala dolazi do divergencije zraka. Pretpostavile smo da stepen divergencije zavisi od hrapavosti površi. Napravile smo simulaciju ove pojave da razmotrimo kriterijume hrapavosti i potvrdimo polaznu pretpostavku. Program koji smo uradile u MATLAB-u dao je vizuelni prikaz distribucije intenziteta svetlosti nakon odbijanja, dok je analiza grafika izvršena u ORIGIN-u. Uočena je zavisnost raspona visina neravnina-izbočina na površi (uzet za osnovni kriterijum hrapavosti) od relativne širine Gausove raspodele intenziteta svetlosti odbijenog snopa (alternativni kriterijum). Dobijeni rezultati otvaraju mogućnosti za eksperimentalnu proveru valjanosti oba razmatrana načina, kao i eventualno određivanje međusobne zavisnosti ovih, a i nekih drugačije definisanih, merljivih parametara hrapavosti.

Uvod

Hrapavost kao karakteristika površine može se definisati na više načina. Često je, međutim, teško izmeriti parametre na mikroskopskom nivou. Zgodno je da se neravnost meri nekom optičkom metodom zbog lakoće i preciznosti. Mi smo došle na ideju da ispitujemo snop odbijen od površi. Ukoliko bi se pokazalo da rezultat na neki način zavisi od hrapavosti, mogli bismo razviti takvu optičku metodu. Kompjuterska simulacija procesa je brz način provere ove pretpostavke koji nam lako daje rezultat za bilo koju usvojenu definiciju.

Neravna površ modelira se prostornom mrežom trougaonih ravnih površi određenom rastojanjima čvorova, temena trouglova, od fiktivne *osnovne ravni,* pri čemu su temena sa iste strane te ravni. Trouglovi su dovoljno malih dimenzija da bi mreža predstavljala dobru aproksimaciju realne hrapave površi (u principu nije ni bitno da se koriste baš trouglovi, ali oni olakšavaju programerski deo posla).

Ivana Medoš (1980), Ruma, D4 B/6, učenica 3. razreda Matematičke gimnazije u Beogradu

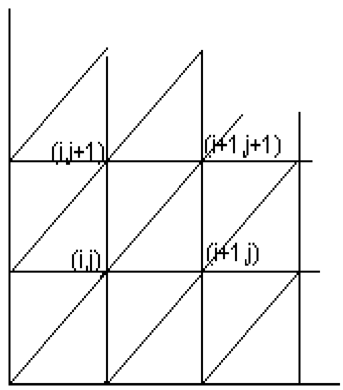
Jelena Pejković (1980), Beograd–Borča, Bratstva–jedinstva 127, Red Cross Nordic United World College, Flekke, Norvege

Distribuciju intenziteta odbijenog zraka smo posmatale na ravni normalnoj na pravac zraka. Očekivale smo da to bude razvučena Gausova raspodela, tj. sa većom relativnom širinom od one kod upadnog snopa. Program koji smo uradile u MATLAB-u dao je vizuelni prikaz distribucije intenziteta svetlosti nakon odbijanja, dok je analiza grafika izvršena u ORIGIN-u.

Realizacija modela

Program u MATLAB-u sastoji se od šest delova. Glavni deo, u kome se definišu sve promenljive i konstante, poziva funkciju *Matrica*, koja, opet, u svom izvršavanju koristi tri funkcije – *Odbijeni*, *TackaNiz* i *StaviNiz* (vidi prilog). Mnogi delovi programa koriste u operacijama sa vektorima koriste i *Proizvod* kao pomoćnu funkciju za izračunavanje vektorskog proizvoda dva zadata vektora.

Tabla, površ od koje se svetlost odbija, predstavljena je matricom. Zadata je realna veličina table. Susedna polja matrice predstavljaju temena trougla, a brojevi dodeljeni poljima matrice rastojanja temena od osnovne ravni. Projekcije trouglova na osnovnu ravan izgledaju kao na slici 1: kvadrati određeni susednim poljima matrice (i, j), (i+1, j), (i, j+1), (i+1, j+1) podeljeni su na dva trougla sa temenima (i, j), (i+1, j), (i+1, j+1) i (i, j), (i, j+1), (i+1, j+1).



Slika 1.
Podela matrice.

Figure 1.
Matrix subdivision.

Matrica je pravljena u MATLAB-u tako da su visine temena trouglova dodeljene na slučajan način, ali je njihova ukupna raspodela Gausova. U programu postoji mogućnost da se raspon visina u matrici podesi tako da su visine reda veličine samih trouglova:

`Tabla=rand(MaxmatrX)/500*E^4;` (glavni deo programa)

`Randn(_)` daje visine iz normalne distribucije širine 1. Množenjem sa bilo kojim koeficijentom, u ovom slučaju $1/(500 \cdot 10^4)$, menja se širina.

Promenom raspona visina menjamo hrapavost table. Dakle, ovde raspon visina usvajamo kao kriterijum hrapavosti.

Upadni snop predstavljen je vektorom čije su koordinate date u odnosu na tablu, pri čemu je *xy* osnovna ravan. Upadni snop pogađa tačku table iznad centra osnovne ravni. Svakom trouglu dodeljujemo zrak koji pogađa njegovo težište pod određenim uglom i odbija se pod tim istim uglom. Intenzitet odbijenog zraka zavisi od intenziteta upadnog zraka (Gausovskog snopa) i površine projekcije trougla na ravan normalnu na upadni zrak. Funkcija *Odbijeni* vraća koordinate vektora odbijenog zraka, dok se intenzitet reguliše funkcijom *Matrica*:

```
intenz=Proizvod(odbV,normalaNN)*exp(-alfa*x).
```

Odbijeni zrak seče 'zaklon' u ravni normalnoj na pravac koji bi imao nakon odbijanja od osnovne ravni.

Zaklon takođe predstavljamo kao matricu koja je izdeljena na kvadratna polja. Zrak odbijen od svakog pojedinačnog trugla pada na jedno od tih polja (*TackaNiz* određuje koordinate tačke u kojoj zrak seče zaklon, ali u sistemu vezanom za zaklon). Zbir intenziteta svih zraka koje padaju u neko polje predstavlja njegovu kumulativnu vrednost (te vrednosti određuje funkcija *StaviNiz* koristeći rezultate funkcije *TackaNiz*).

Aproksimacije

U programu nismo uzele u obzir mogućnost da neki trougao zaklanja druge trouglove, kao i mogućnost da se zrak odbije o nekoliko površi trouglova pre nego što padne na zaklon. Ako ugao pod kojim zrak pada na tablu (u odnosu na osu normalnu na tablu) nije velik, ovi efekti ne utiču značajno na dobijeni rezultat.

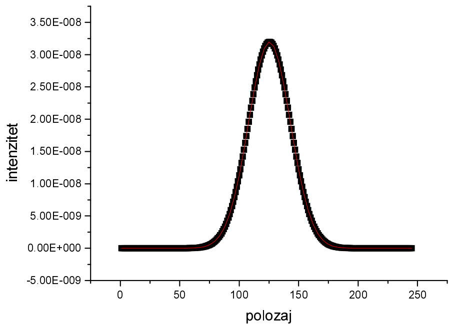
Upadni zraci u našem modelu su paralelni, što je dobra aproksimacija realnog laserskog snopa jer je njegov ugao divergencije mali. Snop svetlosti odbijen od jednog trougla je u obliku trostrane prizme, ali smo ga mi aproksimirali zrakom (linijom). To mnogo olakšava realizaciju modela, a sasvim je prihvatljivo jer su dimenzije trouglova mnogo manje od veličine table.

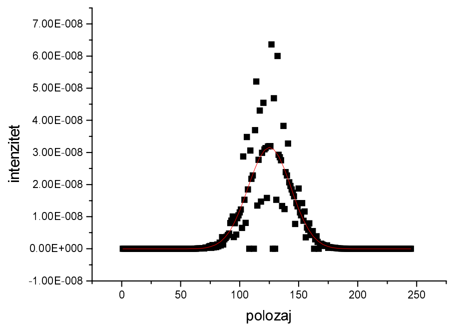
Testiranje modela

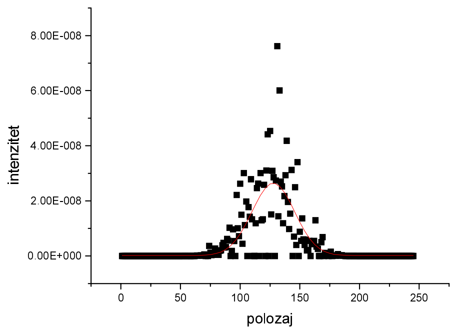
Parametre kao što su dimenzije table i zaklona, upadni ugao snopa, raspodela intenziteta snopa možemo da menjamo i prilagođavamo. Realan proces modelirale smo snopom sa Gausovom raspodelom intenziteta i matricom *Tabla* (vidi prilog). Na ovaj način smo analizirale šest slika odbijene svetlosti za različite hrapavosti.

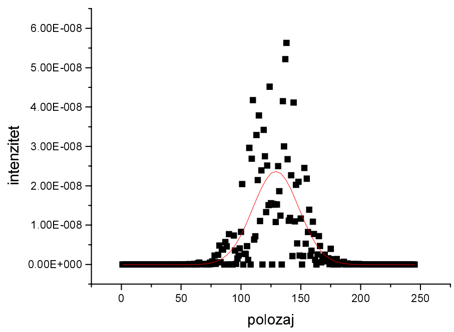
Slika 2 (naspramna strana). Grafik distribucije intenziteta svetlosti duž prave kroz centar snopa:
1. *hrapavost 0 relativna širina snopa 1.000*
2. *hrapavost 1 relativna širina snopa 1.004*
3. *hrapavost 16 relativna širina snopa 1.010*
4. *hrapavost 3, relativna širina snopa 1.087*
5. *hrapavost 64 relativna širina snopa 1.177*
6. *hrapavost 128 relativna širina snopa 2.018.*

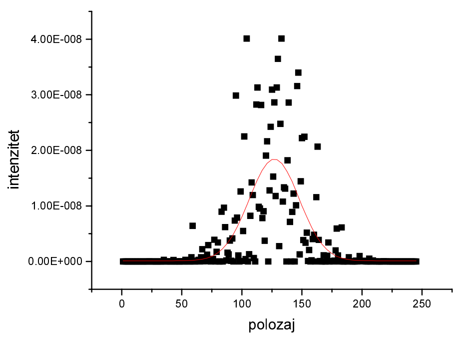
Figure 2 (opposite page). Light intensity distribution along the axis with the beam:
1. *roughness 0, relative beam width 1.000*
2. *roughness 0, relative beam width 1.000*
3. *roughness 0, relative beam width 1.000*
4. *roughness 0, relative beam width 1.000*
5. *roughness 0, relative beam width 1.000*
6. *roughness 0, relative beam width 1.000.*

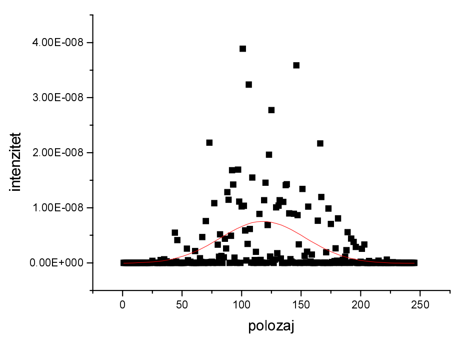

3.50E-008
3.00E-008
2.50E-008
2.00E-008
1.50E-008
1.00E-008
5.00E-009
0.00E+000
-5.00E-009
intenzitet
0
50
100
150
200
250
polozaj

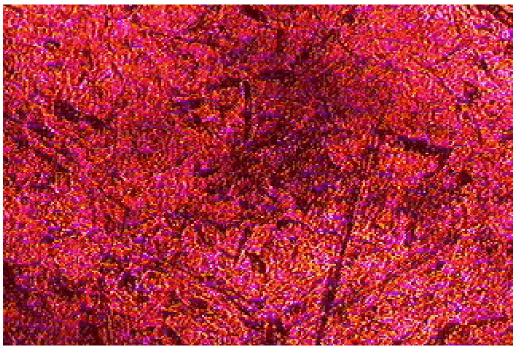

7.00E-008
6.00E-008
5.00E-008
4.00E-008
3.00E-008
2.00E-008
1.00E-008
0.00E+000
-1.00E-008
intenzitet
0
50
100
150
200
250
polozaj

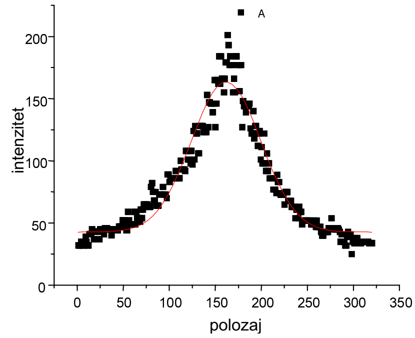

8.00E-008
6.00E-008
4.00E-008
2.00E-008
0.00E+000
intenzitet
0
50
100
150
200
250
polozaj

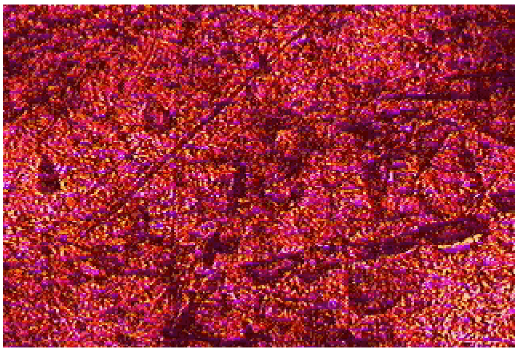

6.00E-008
5.00E-008
4.00E-008
3.00E-008
2.00E-008
1.00E-008
0.00E+000
intenzitet
0
50
100
150
200
250
polozaj

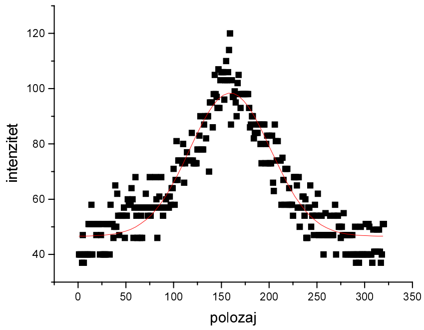

4.00E-008
3.00E-008
2.00E-008
1.00E-008
0.00E+000
intenzitet
0
50
100
150
200
250
polozaj


4.00E-008
3.00E-008
2.00E-008
1.00E-008
0.00E+000
intenzitet
0
50
100
150
200
250
polozaj









Rezultati i diskusija

Rezultati analize odbijene svetlosti za različite hrapavosti prikazani su na slici 1, pri čemu je hrapavost, tj. raspon visina zadat u relativnim jedinicama (veličina slike na zaklonu zavisi od veličine table i rastojanje od zaklona do table). Rezultati su više nego rečiti – da, sa povećanjem hrapavosti zaista raste i stepen divergencije snopa. Što se tiče neke kvantitativne korelacije između dva proučavana kriterijuma hrapavosti, mi za sada ne možemo da uspostavimo teorijski opravdanu vezu. Za to je potreban veliki broj obrađenih slučajeva za što veće dimenzije matrice *Tabla* (time se relativna veličina trouglova u odnosu na *Tablu* smanjuje, a 'finoća' povećava), i to na nekom programskom jeziku nižeg nivoa. MATLAB, koji smo izabrale zbog grafičkih mogućnosti, suviše je spor za takav posao.

Bilo bi zanimljivo uporediti simulirani model i realni fizički sistem. Mi smo uradile malu eksperimentalnu proveru. Metalnu površ izlagale smo dejstvu kiseline, a potom snimale i analizirale laserske snopove odbijene od te površi u različitim stadijumima nagriženosti (slika 2). Ali (opet), potrebno je mnogo više merenja da bi se odredila tražena zavisnost, na osnovu koje bi se mogla uspostaviti veza između novog kriterijuma, vremena dejstva kiseline, i onog osnovnog u našem modelu – raspona i visina. Bilo bi dobro razmotriti i neke druge kriterijume. Mi smo pošle od pretpostavke da kiselina povećava raspon visina neravnina, ali to nije provereno i verovatno važi samo do određenog trenutka procesa nagrizanja. Jedan od alternativnih načina je da se za meru hrapavosti usvoji odnos površina neravne (nagrižene) i ravne površi (pre oštećenja). Postoje neki metodi za eksperimentalno određivanje ove veličine, a za naš model ona sasvim lako može da se izračuna. Do sada dobijeni rezultati ukazuju na niz novih mogućnosti za definisanje hrapavosti i njeno povezivanje sa opisanim svetlosnim efektom.

Slika 3 (naspramna strana). Snimci novčića CCD kamerom i odgovarajući grafici raspodele

gore: vreme dejstva kiseline 0, relativna širina raspodele 1.000

dole: vreme dejstva kiseline 12 h, relativna širina raspodele 1.130

Figure 3 (opsite page). CCD shots of coins and corresponding distribution graphs

up: duration of acid treatment 0, relative distribution width 1.000

down: duration of acid treatment 12h, relative distribution width 1.130

Literatura

Wyant, J. C. 1998. *Optics source book – Speckle*. McGraw-Hill.

Ćalasan, L., Petrovska, M. 1995. *MATLAB i dodatni moduli Control System Toolbox i SIMULINK*. Beograd: Mikro knjiga.

Ivana Medoš and Jelena Pejković

Width of Gaussian Distribution of Light Intensity Reflected from a Rough Metal Surface as a Measure of Roughness

Roughness as characteristic can be defined in many ways. However, its often hard to determine its parameters on the microscopic scale. An optical method would be more convenient for measuring because of its interaction. After reflection light rays diverge, and the normal (Gaussian) distribution of intensity grows wider. If the divergence depended on the level of roughness, we could develop the optical method. A computer simulation of the process is a quick way of checking this assumption for any accepted definition of roughness.

In MATLAB we modeled a rough surface as a 3D net of plane triangles determined by the distances of its vertices from a fictive *basic plane*, the vertices being on the same side of that plane. Triangles are small enough to make the net – a good approximation of a real rough surface (not necessarily should the parts of a modeled surface by triangles, but this way the programming is much facilitated). The net is described by a matrix whose numbers are heights of the triangles. The numbers are

randomly chosen, but in such manner that their general distribution is Gaussian (by function *randn* in MATLAB). We can fit them to be on the same order of magnitude as the triangles. Now roughness can be defined as span of the heights. We programmed initial and reflected beams of light as vectors in space, and viewed the intensities of the reflected ones in a plane normal to it, along the line through its center.

Six graphs done in ORIGIN, analysing results of our simulation, gave undoubtful answer – yes, there is a connection between roughness and divergence. The larger is the span of the heights, the wider is the corresponding Gaussian curve (see the figure 2; relative width is division of widths of any curve and the curve of the light reflected from a leveled surface, when roughness is 0). However, for determining any correlation between the two criteria, we need larger number of analyzed cases and results for matrices of larger dimensions (when the dimensions of the triangles are smaller comparing to the entire surface). MATLAB, which we chose for its graphic facilities, is unfortunately too slow for such a job. It would be also interesting to compare the simulation with a real physics system. Actually, we have done a small experimental test. By CCD we shot images of both a metal surface before and after treatment with an acid, and light beams reflected from it. The experiment just confirmed our initial assumption (Figure 3). Our work also opens possibilities of defining roughness and finding out their relationship with the light effect.

Prilog: listing programa

```
format long;

global MaxmatrX; MaxmatrX=100;
global MaxmatrY; MaxmatrY=MaxmatrX; % KVADRATNA TABLA
global DuzinaTableX; DuzinaTableX=0.01;
global DuzinaTableY; DuzinaTableY=0.01;

global DuzinaZaklonaX; DuzinaZaklonaX=0.01;
global DuzinaZaklonaY; DuzinaZaklonaY=0.01;

global alfa; alfa=1e6;

global K0;
global K1;
global N0;
global L;

% Unos Kordinata vektora upadnog snopa
% ----------------------------------
K0(1)=0;
K0(2)=0;
K0(3)=-1;
K0

% Unos Normala na tablu
% --------------------
N0(1)=0;
N0(2)=0;
N0(3)=1;
N0
N0=N0./norm(N0);

% Unos Rastojanje zaklona od table
% -------------------------------
L=0.05

lambda=- Proizvod(N0,K0);
K1(1)=lambda*2*N0(1)+K0(1);
K1(2)=lambda*2*N0(2)+K0(2);
K1(3)=lambda*2*N0(3)+K0(3);

% Generisanje table
% ----------------
global Tabla;

Tabla= zeros(MaxmatrX);
% hrapava tabla se formira OVDE
Tabla=randn(MaxmatrX)/500e4;
%for i=1:MaxmatrX
%       for j=1:MaxmatrY
```

```
%    Tabla(i,j)=sin(i/4)*cos(j/4)*10e-7;
%end
%end

global CentarTable;
   CentarTable(1)=DuzinaTableX/2;
   CentarTable(2)=DuzinaTableY/2;
   CentarTable(3)=0;
global intK1;
   intK1= norm(K1)
global EK1;
   EK1= K1./intK1;
global Zaklon; Zaklon=zeros(MaxmatrX);
global Xvek Yvek i j;
   Xvek(1)=-EK1(3);
   Xvek(2)=0;
   Xvek(3)=EK1(1);
   Xvek=Xvek./norm(Xvek);

   Yvek(1)=EK1(2)*Xvek(3)-EK1(3)*Xvek(2);
   Yvek(2)=EK1(3)*Xvek(1)-EK1(1)*Xvek(3);
   Yvek(3)=EK1(1)*Xvek(2)-EK1(2)*Xvek(1);
   Yvek=Yvek./norm(Yvek);


SHT=Matrica;

global SHT; SHT=0;

function y=matrica

     intenz=0;
     global MaxmatrX MaxmatrY
     global Tabla
     global DuzinaTableX DuzinaTableY
     global CentarTable
     global K0
     global alfa
     global SHT

     for i=1:MaxmatrX-1
         i
         for j=1:MaxmatrY-1
             for k=1:2
                 a(1)=DuzinaTableX/MaxmatrX*i;
                 a(2)=DuzinaTableY/MaxmatrY*j;
                 a(3)=Tabla(i,j);
                 if k==1
                   b(1)=DuzinaTableX/MaxmatrX*(i+1);
                   b(2)=DuzinaTableY/MaxmatrY*(j+1);
                   b(3)=Tabla(i+1,j+1);

                   c(1)=DuzinaTableX/MaxmatrX*i;
                   c(2)=DuzinaTableY/MaxmatrY*(j+1);
```

```
                c(3)=Tabla(i,j+1);
           else
              b(1)=DuzinaTableX/MaxmatrX*(i+1);
              b(2)=DuzinaTableY/MaxmatrY*j;
              b(3)=Tabla(i+1,j);

              c(1)=DuzinaTableX/MaxmatrX*(i+1);
              c(2)=DuzinaTableY/MaxmatrY*(j+1);
              c(3)=Tabla(i+1,j+1);
           end

          [odbV, normalaNN, imaO, intenz]=odbi-
  jeni(a,b,c);

        if imaO
           teziste(1)=( a(1)+b(1)+c(1) )/3;
           teziste(2)=( a(2)+b(2)+c(2) )/3;
           teziste(3)=( a(3)+b(3)+c(3) )/3;

          [tacka, imaT]= TackaNaZ(odbV, teziste);
           CT(1)=-CentarTable(1)+teziste(1);
           CT(2)=-CentarTable(2)+teziste(2);
           CT(3)=-CentarTable(3)+teziste(3);

           K0norm= K0./norm(K0);

           x=CT(1)*CT(1)+CT(2)*CT(2)+CT(3)*CT(3)-
  Proizvod(CT,K0norm)^2;

                      intenz=Proizvod(odbV,nor-
  malaNN)*exp(-alfa*x);
 if imaT
             SHT= StaviNaZ(tacka, intenz);
             end
          end
      end
    end
  end

y=0;

function y=Proizvod(a,b)
     y=a(1)*b(1)+a(2)*b(2)+a(3)*b(3);

function [odbV, normalaNN, ima_odb, intenz]= odbi-
     jeni(a,b,c)

% napravi vektore trougla, vektor normale i odbijeni
     vektor

     vek1(1)=-a(1)+b(1);
     vek1(2)=-a(2)+b(2);
     vek1(3)=-a(3)+b(3);
```

```
    vek2(1)=-a(1)+c(1);
   vek2(2)=-a(2)+c(2);
   vek2(3)=-a(3)+c(3);

   normalaNN(1)= vek1(2)*vek2(3) - vek2(2)* vek1(3);
   normalaNN(2)= -vek1(1)*vek2(3) + vek2(1)* vek1(3);
   normalaNN(3)= vek1(1)*vek2(2) - vek2(1)*vek1(2);

   intenz= norm(normalaNN);
   normala= normalaNN./intenz;


% odbijeni vektor BEZ EFEKTA POVRSINE TROUGLA
% ------------------------------------------

 global K0;

   lambda=-Proizvod(normala,K0);
   odbV(1)=lambda*2*normala(1)+K0(1);
   odbV(2)=lambda*2*normala(2)+K0(2);
   odbV(3)=lambda*2*normala(3)+K0(3);

   ima_odb= lambda0;


function [tacka, imaT]=TackaNaZ(odbV, teziste)

   global EK1;
   global CentarTable;
   global L;

tacka(1)=0; tacka(2)=0; tacka(3)=0;

   imaT= ~(Proizvod(odbV, EK1)==0);

   if imaT==1
        mi= (  Proizvod(CentarTable,EK1)+L-Proiz-
     vod(teziste,EK1) )/Proizvod(odbV,EK1);

      imaT= mi0;

     if imaT==1
         tacka(1)= teziste(1)+mi*odbV(1);
         tacka(2)= teziste(2)+mi*odbV(2);
         tacka(3)= teziste(3)+mi*odbV(3);
      end

   end
% vrati tacku u odnosu na zaklon
% ----------------------------

tacka(1)= tacka(1)- L*EK1(1)- CentarTable(1);
tacka(2)= tacka(2)- L*EK1(2)- CentarTable(2);
tacka(3)= tacka(3)- L*EK1(3)- CentarTable(3);
```

```
function y=StaviNaZ(tacka, intenz)

     global Zaklon
     global MaxmatrX MaxmatrY
     global DuzinaZaklonaX DuzinaZaklonaY

      zX=floor((tacka(1)+DuzinaZaklonaX/2)*MaxmatrX/
     DuzinaZaklonaX)+1;
      zY=floor((tacka(2)+DuzinaZaklonaY/2)*MaxmatrY/
     DuzinaZaklonaY) +1;

     if (zX0) & (zXaxmatrX) & (zY0) & (zYaxmatrY)
        Zaklon(zX,zY)=Zaklon(zX,zY)+ intenz;
     end

     y=1;
```