

Listings

1	Helpon aloituksen koodi	4
2	Tiedetään tason kolme pistettä, lasketaan yhtälön parametrit	6
3	Tiedetään tason piste ja kaksi suuntavektoria, lasketaan tason yhtälön parametrit	7
4	Tiedetään tason piste ja normaali, lasketaan tason yhtälön parametrit	8
5	Tiedetään tason yhtälön parametrit, piirretään taso	9
6	Tallennetaan kuva muodossa .png	10
7	Tallennetaan kuva muodossa .pdf	11
8	Lasketaan ja piirretään tason origoa lähin piste	14
9	Piirretään tason kuvaan suuntavektorit ja normaalivektori	15
10	Piirretään tason kuvaan suuntavektorit ja normaalivektori	16
11	Piirretään tason ja koordinaattiakselien leikkauspisteet	19
12	Piirretään tason ja koordinaattiakselien leikkauspisteet	20
13	Pyöritetään piirrettyä tasoa	23
14	Pyöritetään piirrettyä tasoa	24
15	Tallennetaan syntynyt video	25
16	Tallennetaan syntynyt video	26

Sisältö

1	Esimerkki: Vaakasuora taso	12
2	Esimerkki: taso, yksikkönormaali ja yksikkösuuntavektorit	17
3	Esimerkki: taso sekä tason ja koordinaattiakselien leikkauspisteet	21
4	Esimerkki: taso ja pyöritysvideon tekeminen siitä	27

Johdanto

Tähän tiedostoon on koottu eräitä MATLAB-koodeja, joilla voi piirtää kuvia tasoista. Ajatuksena on piirtää hienoja kuvia, mutta pitää koodi silti yksinkertaisena.

Toivottavasti voit käyttää MATLABia tai Oktavea ja kokeilla koodeja itse. Voit sitten opetella muokkaamaan koodeja niin, että saat laskettua sinua kiinnostavia esimerkkejä.

Valitettavasti tässä vaiheessa

- koodeja ei ole vielä testattu Oktavessa
- jos haluat piirtää useamman tason samaan kuvaan, saat opetella itse

Jos avaat MATLABin, copypastetat allaolevan koodin ja painat Enteriä, 3d-kuvan tasosta pitäisi tupsahtaa ruutuun. Toivottavasti koodi toimii. Koodi on liitetty tähän “verbatim”-muodossa, pelkkänä tekstinä.

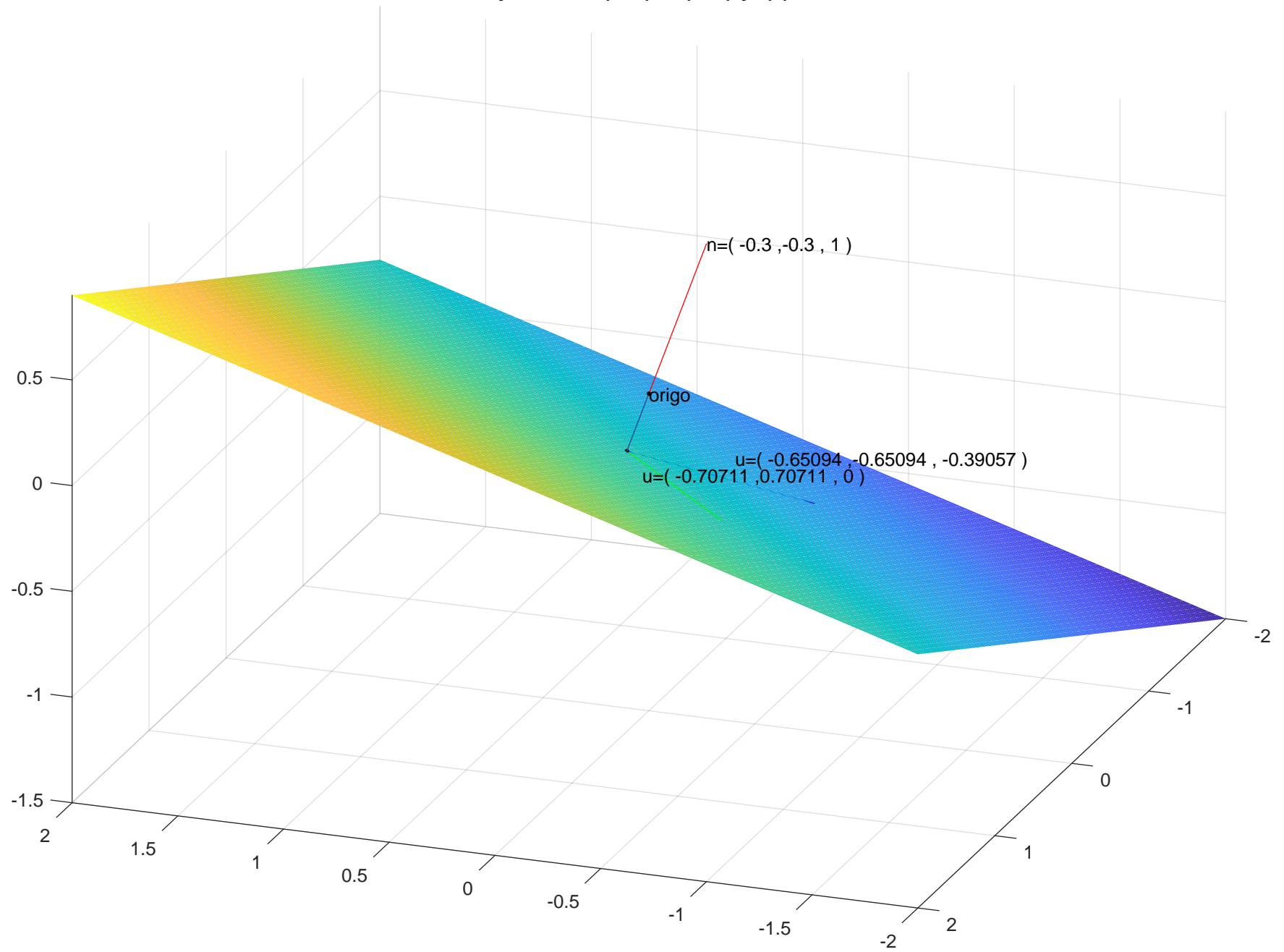
```
mkdir 'koodeja'           %tehdään kansio "koodeja"  
cd 'koodeja'             %siirrytään kansion "koodeja"  
websave('zippi', 'https://integraali.com/vektorilaskenta/matlab/tasonpiirtaminenkoodit.zip');  
                          %ladattiin zip-tiedosto internetistä  
unzip('zippi.zip')      %puretaan zip-tiedosto  
cd 'tasonpiirtaminenkoodit' %siirrytään kansioon tasonpiirtaminenkoodit  
tasoexamples4           %suoritetaan tasoexamples4, joka piirtää kuvan
```

Jatkossa koodit ovat “listing”-muodossa “matlab-prettifier”-paketilla tyylliteltynä. Rivinumerot ja tyyllittelyt helpottavat koodin lukemista. Tällöin copypastetus ei kuitenkaan onnistu. Koodit löytyvät pelkkänä tekstinä osoitteesta integraali.com/vektorilaskenta/matlab/tasonpiirtaminenkoodit.zip.

Listing 1: Helpon aloituksen koodi

```
1 %Piirretään tasoja MATLABilla helposti.  
2 %  
3 %Avaa MATLAB  
4 %kopioi allaoleva koodi MATLABiin ja paina Enteriä  
5 %yhden 3d-kuvan pitäisi ilmestyä ruudulle  
6 mkdir 'koodeja'           %tehdään kansio koodeja  
7 cd 'koodeja'             %siirrytään kansion koodeja  
8 websave('zippi', 'https://integraali.com/vektorilaskenta/matlab/tasonpiirtaminenkoodit.zip');  
9                          %ladattiin zip-tiedosto internetistä  
10 unzip('zippi.zip')      %puretaan zip-tiedosto  
11 cd 'tasonpiirtaminenkoodit' %siirrytään kansioon tasonpiirtaminenkoodit  
12 tasoexamples4           %suoritetaan tasoexamples4, joka piirtää kuvan
```

Tason yhtalo on: $(-0.3)*x+(-0.3)*y+(1)*z=-0.3$



Tiedetään tason kolme pistettä, lasketaan yhtälön parametrit

Listing 2: Tiedetään tason kolme pistettä, lasketaan yhtälön parametrit

```
1 function [a,b,c,d] = tasopqr2abcd(p,q,r)
2 %Laskee tason pisteistä p,q,r
3 %tason yhtälön ax+by+cz=d parametrit a,b,c,d.
4 %
5 %Esimerkiksi
6 %p=[2,0,0];q=[0,3,0];r=[0,0,-1];
7 %[a,b,c,d]=tasopqr2abcd(p,q,r);
8 %
9 %tuottaa
10 %[a,b,c,d]=[-3,-2,6,-6].
11
12 u=q-p;
13 v=r-p;
14 n=cross(u,v);
15 a=n(1);b=n(2);c=n(3);
16 d=n*p';
17 end
```

Tiedetään tason piste ja kaksi suuntavektoria, lasketaan tason yhtälön parametrit

Listing 3: Tiedetään tason piste ja kaksi suuntavektoria, lasketaan tason yhtälön parametrit

```
1 function [a,b,c,d] = tasopuv2abcd(p,u,v)
2 %Laskee tason pisteestä p ja suuntavektoreista u,v
3 %tason yhtälön ax+by+cz=d parametrit a,b,c,d.
4 %
5 %Esimerkiksi
6 %p=[2,0,0];u=[-2,3,0];v=[-2,0,-1];
7 %[a,b,c,d] = tasopuv2abcd(p,u,v);
8 %
9 %tuottaa
10 %[a,b,c,d]=[-3,-2,6,-6].
11
12 n=cross(u,v);
13 a=n(1);b=n(2);c=n(3);
14 d=n*p';
15 end
```

Tiedetään tason piste ja normaali, lasketaan tason yhtälön parametrit

Listing 4: Tiedetään tason piste ja normaali, lasketaan tason yhtälön parametrit

```
1 function [a,b,c,d] = tasopn2abcd(p,n)
2 %Laskee tason pisteestä p ja normaalista n
3 %tason yhtälön ax+by+cz=d parametrit a,b,c,d.
4 %
5 %Esimerkiksi
6 %p=[2,0,0];n=[-3,-2,6];
7 %[a,b,c,d] = tasopn2abcd(p,n);
8 %
9 %tuottaa
10 %[a,b,c,d]=[-3,-2,6,-6].
11
12 a=n(1);b=n(2);c=n(3);
13 d=n*p';
14 end
```


Tiedetään tason yhtälön parametrit, piirretään taso

Listing 5: Tiedetään tason yhtälön parametrit, piirretään taso

```
1 function [h] =tasoabcd2kuva(a,b,c,d,t)
2 %Piirtää kuvan tasosta ax+by+cz=d.
3 %
4 %Esimerkiksi
5 %t=linspace(-2,2,100);
6 %tasoabcd2kuva(-3,-2,6,-6,t);
7
8 [x,y]=meshgrid(t);
9
10 if c~=0,           %lasketaan z, meshgridillä x,y
11 cz=d-a*x-b*y;z=cz/c;
12 elseif b~=0,      %lasketaan y, meshgridillä x,z
13 z=y; by=d-a*x;y=by/b;
14 elseif a~=0,      %lasketaan x, meshgridillä y,z
15 z=x; x=ones(size(y))*d/a;
16 else disp('Virhe! Tason normaali ei voi olla nollavektori.')
17 end
18
19 figure('units','normalized','outerposition',[0 0 1 1])
20 h=surf(x,y,z);
21 set(h,'linestyle','none')
22 axis equal, hold on
23 end
```

Tallennetaan kuva muodossa .png

Listing 6: Tallennetaan kuva muodossa .png

```
1 function [] =kuva2png(nimi)
2 %Tallentaa kuvan png-tiedostona. Huolehdi siitä, että kuva, jonka haluat
3 %tallentaa, on aktiivinen.
4 %
5 %Esimerkiksi
6 %
7 %x=linspace(-1,1,100);y=x.^2;
8 %figure('units','normalized','outerposition',[0 0 1 1])
9 %plot(x,y),axis equal,title('Funktio $y=x^2$', 'Interpreter', 'Latex', 'FontSize', 30)
10 %kuva2png() %tai kuva2png('jokin-tiedostonimi')
11
12 h=gcf;
13 %set(h,'PaperOrientation','portrait');
14 %set(h,'PaperUnits','normalized');
15 %set(h,'PaperPosition', [0 0 1 1]);
16
17 if nargin==1, knimi=nimi;
18 else
19 %Kuvan nimeksi voidaan asettaa kellonaika
20 knimi=['kuva',num2str(-floor(clock))]; %poimitaan kellonaika
21 knimi=knimi(knimi~=' '); %poistetaan välilyönnit
22 end
23 print(h, '-dpng', knimi);
24 %print(h, '-djpg', 'knimi');
25 end
```

Tallennetaan kuva muodossa .pdf

Listing 7: Tallennetaan kuva muodossa .pdf

```
1 function [] =kuva2pdf(nimi)
2 %Tallentaa kuvan pdf-tiedostona. Huolehdi siitä, että kuva, jonka haluat
3 %tallentaa, on aktiivinen.
4 %
5 %Esimerkiksi
6 %
7 %x=linspace(-1,1,100);y=x.^2;
8 %figure('units','normalized','outerposition',[0 0 1 1])
9 %plot(x,y),axis equal,title('Funktio $y=x^2$', 'Interpreter', 'Latex', 'FontSize', 30)
10 %kuva2pdf() %tai kuva2pdf('jokin-tiedostonimi')
11
12 h=gcf;
13 set(h, 'PaperOrientation', 'landscape');
14 set(h, 'PaperUnits', 'normalized');
15 set(h, 'PaperPosition', [0 0 1 1]);
16
17 if nargin==1, knimi=nimi;
18 else
19 %Kuvan nimeksi voidaan asettaa kellonaika
20 knimi=['kuva', num2str(-floor(clock))]; %poimitaan kellonaika
21 knimi=knimi(knimi~=' '); %poistetaan välilyönnit
22 end
23 print(h, '-dpdf', knimi);
24 %print(h, '-dpng', 'knimi');
25 end
```

1 Esimerkki: Vaakasuora taso

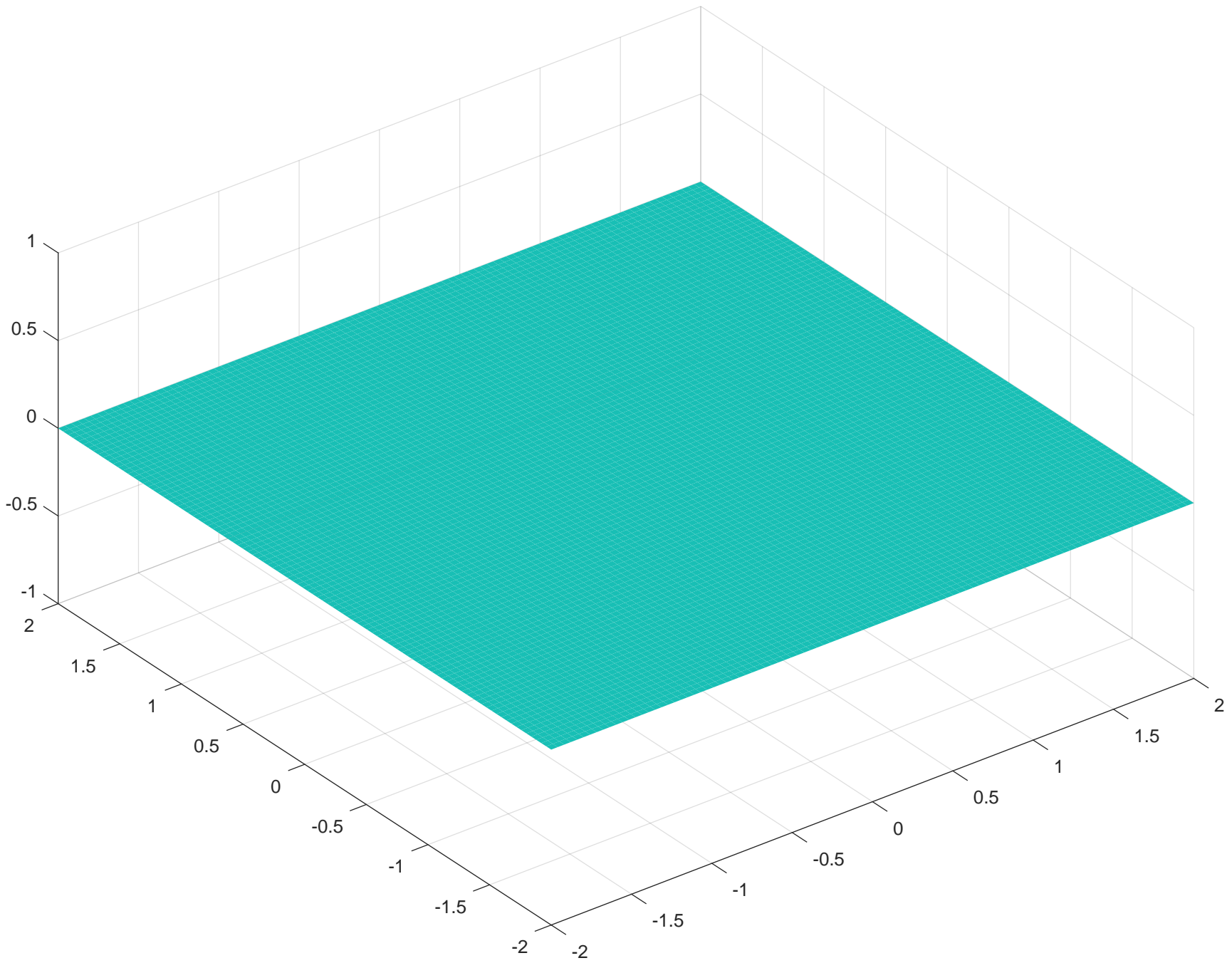
Halutaan piirtää yksinkertaisia kuvia tasoista. Piirretään esimerkiksi vaakasuora taso $z = 0$.

Tason $z = 0$ yhtälön parametrit ovat $a = 0, b = 0, c = 1, d = 0$.

Ajetaan siis seuraava koodi, jonka voit halutessasi copypastettaa MATLABiin

```
a=0;b=0;c=1;d=0;  
t=linspace(-2,2,100);  
tasoabcd2kuva(a,b,c,d,t);
```

Kuva on seuraavalla sivulla. Selvästi kuva on vaakasuorasta tasosta $z = 0$.



Lasketaan ja piirretään tason origoa lähin piste

Listing 8: Lasketaan ja piirretään tason origoa lähin piste

```
1 function [olahin] =tasonorigoalahin(a,b,c,d)
2 %Laskee ja piirtää tason origoa lähimmän pisteen.
3 %
4 %Esimerkiksi
5 %a=-3;b=-2;c=6;d=-6;
6 %[olahin] =tasonorigoalahin(a,b,c,d)
7 %
8 %tuottaa
9 %olahin = 0.3673, 0.2449,-0.7347
10
11 n=[a,b,c];T=norm(n)^2;
12 olahin=[a,b,c]*d/T; %origoa lähin piste
13 oolahin=[0,0,0;olahin]';
14 plot3(oolahin(1,:),oolahin(2,:),oolahin(3,:))
15 plot3(oolahin(1,:),oolahin(2,:),oolahin(3,:), 'k.', 'LineWidth',20)
16 text(0,0,0, 'origo')
17 %text(olahin(1),olahin(2),olahin(3),'origoa lähin piste')
18 end
```

Piirretään tason kuvaan suuntavektorit ja normaalivektori

Listing 9: Piirretään tason kuvaan suuntavektorit ja normaalivektori

```
1 function [u,v,n] =tasokuvauvn(a,b,c,d)
2 %Piirtää tason kuvaan suuntavektorit ja normaalivektorin
3 %
4 %Esimerkiksi
5 %tasokuvauvn(-3,-2,6,-6)
6
7 %Lasketaan origoa lähin piste ja asetetaan vektorit alkamaan siitä
8 n=[a,b,c];T=norm(n)^2;
9 olahin=[a,b,c]*d/T;          %origoa lähin piste
10 %Lasketaan normaali ja kaksi tason suuntavektoria
11 sn=n/norm(n);              %yksikkönormaali
12 su=[0,0,0];
13 su=[b-c,c-a,a-b];
14 if norm(su)==0,
15 su=[-b+c,c+a,-a-b];
16 end
17 su=su/norm(su);            %suuntavektori u
18 sv=cross(sn,su);          %suuntavektori v
```

Listing 10: Piirretään tason kuvaan suuntavektorit ja normaalivektori

```
19
20 %Piirretään n,u,v
21 osn=[olahin;olahin+sn]';
22 osu=[olahin;olahin+su]';
23 osv=[olahin;olahin+sv]';
24 plot3(osn(1,:),osn(2,:),osn(3,:), 'r')
25 plot3(osu(1,:),osu(2,:),osu(3,:), 'g')
26 plot3(osv(1,:),osv(2,:),osv(3,:), 'b')
27 %tekstit vektorien u ja v päätepisteiden yläpuolelle
28 ute=osu(:,end)+0.3*sn;
29 vte=osv(:,end)+0.3*sn;
30 %Pidemmät tekstit lukuarvoilla
31 text(osn(1,end),osn(2,end),osn(3,end), ['n=( ' num2str(a), ' , ' ,num2str(b), ' , ' ,num2str(c), ' )'])
32 text(ute(1,end),ute(2,end),ute(3,end), ['u=( ' num2str(su(1)), ' , ' ,num2str(su(2)), ' , ' ,num2str(su
    (3)), ' )'])
33 text(vte(1,end),vte(2,end),vte(3,end), ['u=( ' num2str(sv(1)), ' , ' ,num2str(sv(2)), ' , ' ,num2str(sv
    (3)), ' )'])
34 title(['Tason yhtälö on: ( ' , num2str(a), ' ) *x+( ' ,num2str(b), ' ) *y+( ' ,num2str(c), ' ) *z=' , num2str(d)])
35 end
```


2 Esimerkki: taso, yksikkönormaali ja yksikkösuuntavektorit

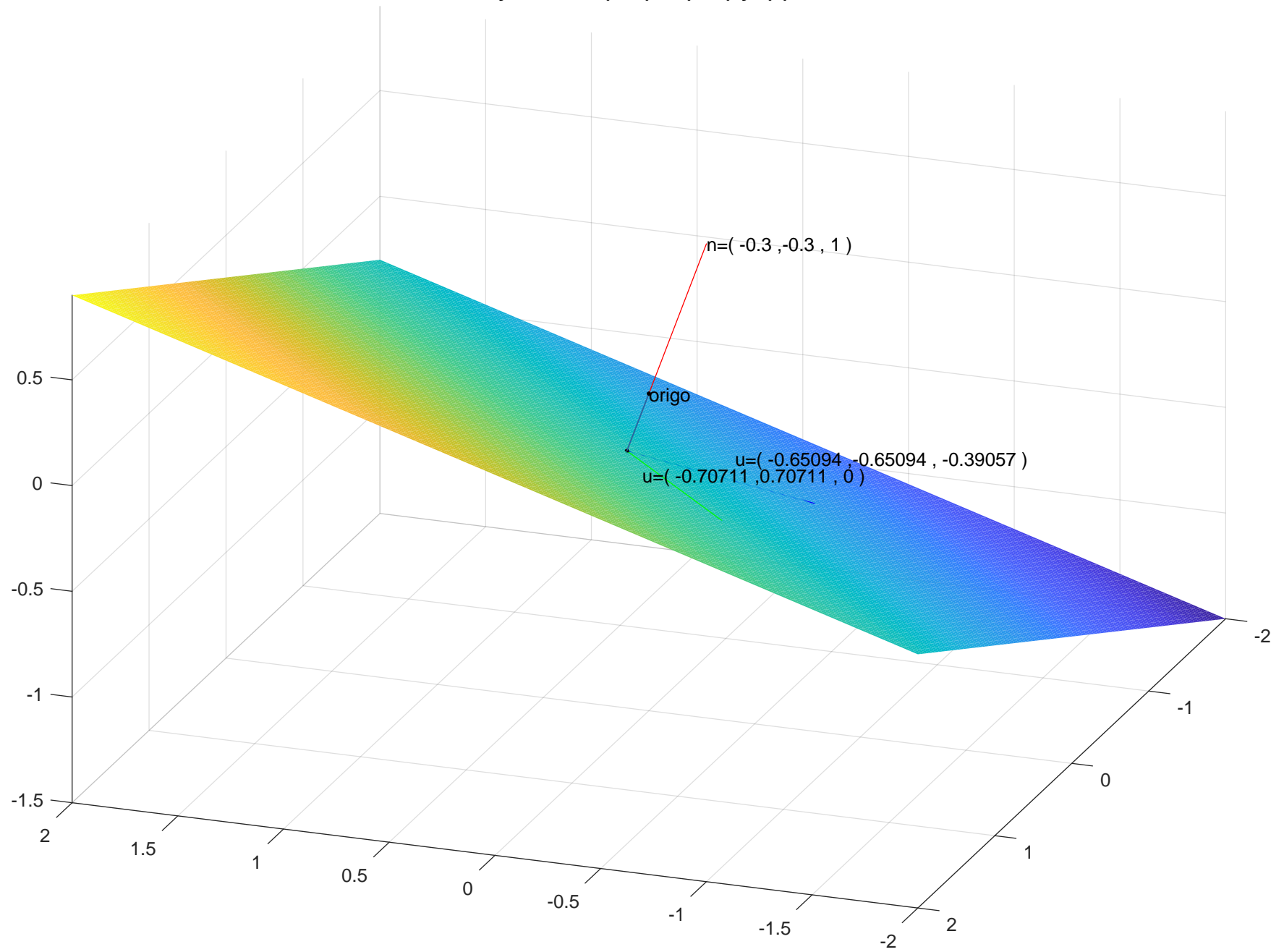
Halutaan piirtää taso, tason yksikkönormaali ja tason yksikkösuuntavektorit. Valitaan tason pisteiksi $p = (1, 0, 0)$, $q = (0, 1, 0)$ ja $r = (0, 0, 0.3)$.

Ajetaan siis seuraava koodi, jonka voit halutessasi copypastettaa MATLABiin

```
%taso, yksikkönormaali, yksikkösuuntavektorit
p=[1 0 0];q=[0,1,0];r=[0,0,-0.3]; %määritellään pisteet
[a,b,c,d]=tasopqr2abcd(p,q,r); %tason yhtälön parametrit
t=linspace(-2,2,100); %olkoot x,y välillä [-2,2]
tasoabcd2kuva(a,b,c,d,t); %piirretään taso
[olahin] =tasonorigoalahin(a,b,c,d);%piirretään origoa lähin piste
tasokuvauvn(a,b,c,d) %piirretään suuntavektorit ja normaali
view(200,20) %tarkastellaan suunnasta 200,20
kuva2png() %tallennetaan .png
```

Kuva on seuraavalla sivulla. Esimerkiksi tason normaali on $n = (-0.3, -0.3, 1)$. Normaalin komponenteista z -komponentti on suurin. Normaali osoittaa siis “melkein” z -akselin suuntaan ja taso on “melkein” vaakasuora.

Tason yhtalo on: $(-0.3)*x+(-0.3)*y+(1)*z=-0.3$



Piirretään tason ja koordinaattiakselien leikkauspisteet

Listing 11: Piirretään tason ja koordinaattiakselien leikkauspisteet

```
1 function [] = taso8akseli(a,b,c,d)
2 %Tason ja akselien leikkauspisteet.
3 %
4 %Esimerkiksi
5 %taso8akseli(-3,-2,6,-6)
6
7 taso8x=[0,0,0];
8 taso8y=[0,0,0];
9 taso8z=[0,0,0];
10
11 if a~=0,
12 taso8x=[d/a,0,0];
13 plot3(d/a,0,0,'k.','LineWidth',20),
14 plot3([0,2*d/a],[0,0],[0,0])
15 text(taso8x(1),taso8x(2),taso8x(3),['x=( ' num2str(taso8x(1)), ' , ',num2str(taso8x(2)), ' , ',num2str
    (taso8x(3)), ' )'])
16 end
17 if b~=0,
18 taso8y=[0,d/b,0];
19 plot3(0,d/b,0,'k.','LineWidth',20),
20 plot3([0,0],[0,2*d/b],[0,0])
21 text(taso8y(1),taso8y(2),taso8y(3),['y=( ' num2str(taso8y(1)), ' , ',num2str(taso8y(2)), ' , ',num2str
    (taso8y(3)), ' )'])
22 end
```

Listing 12: Piirretään tason ja koordinaattiakselien leikkauspisteet

```
23 if c~=0,
24   taso8z=[0,0,d/c];
25   plot3(0,0,d/c,'k.','LineWidth',20),
26   plot3([0,0],[0,0],[0,2*d/c])
27   text(taso8z(1),taso8z(2),taso8z(3),['z=( ' num2str(taso8z(1)), ' , ',num2str(taso8z(2)), ' , ',num2str
      (taso8z(3)), ' )'])
28 end
29
30 %Leikkauspistefunktion nimi on väännös vitsistä ''8''=''eight''=''ate''.
31 %
32 %Q: Why was 9 afraid of 7?
33 %A: Because 7,8,9.
```

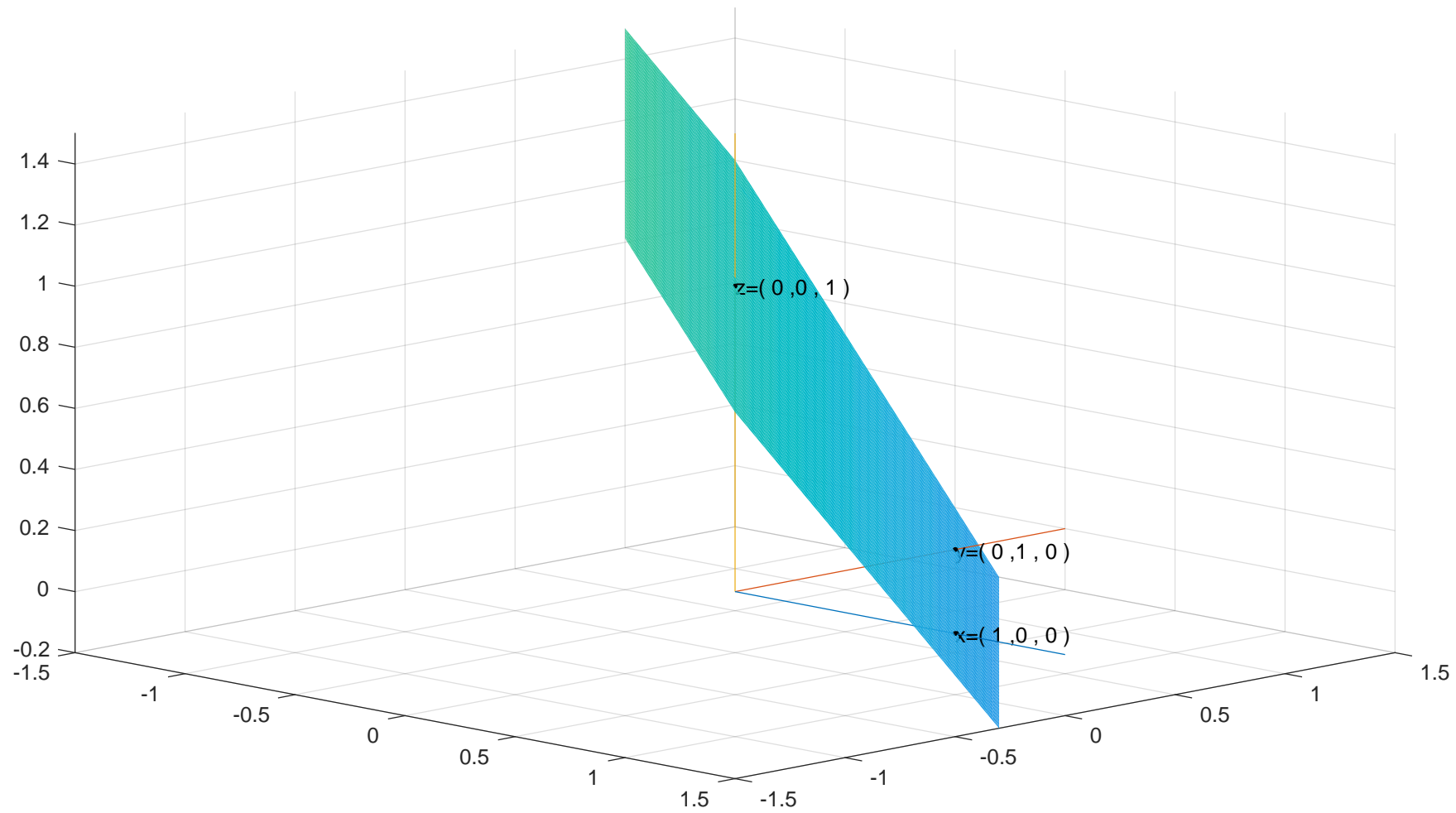
3 Esimerkki: taso sekä tason ja koordinaattiakselien leikkauspisteet

Halutaan piirtää taso ja löytää, missä pisteissä taso leikkaa koordinaattiakselit. Valitaan tason yhtälöksi $x + y + z = 1$. Tällöin pisteet $(1, 0, 0)$, $(0, 1, 0)$ ja $r = (0, 0, 1)$ kuuluvat tasoon.

Ajetaan siis seuraava koodi, jonka voit halutessasi copypastettaa MATLABiin

```
t=linspace(-2,2,100);  
a=1;b=1;c=1;d=1;  
tasoabcd2kuva(a,b,c,d,t);  
taso8akseli(a,b,c,d)  
axis([-1.5 1.5 -1.5 1.5 -0.2 1.5])  
view(45,11)  
kuva2pdf()
```

Kuva on seuraavalla sivulla.



Pyöritetään piirrettyä tasoa

Listing 13: Pyöritetään piirrettyä tasoa

```
1 function [f] =videopyor(kulma, korkeus)
2 %Pyörittää kuvaa ja nauhoittaa videon
3 %
4 %Esimerkiksi
5 %
6 %s=linspace(-2,2,100);
7 %[x,y]=meshgrid(s);
8 %z=x.^2+y.^2;
9 %figure('units','normalized','outerposition',[0 0 1 1])
10 %h=surf(x,y,z);
11 %set(h,'linestyle','none')
12 %axis equal, hold on
13 %kulma=linspace(1,360,360);
14 %korkeus=20+90*sin(kulma/180);
15 %f=videopyor(kulma, korkeus);
16 %%f=videopyor(kulma,20) %vakiokorkeus
17
18 N=length(kulma);
19
20 if numel(korkeus)==1,
21     korkeus=korkeus*ones(N,1);
22 end
```

Listing 14: Pyöritetään piirrettyä tasoa

```
23
24 if numel(kulma)~=numel(korkeus),
25     disp('Vektoreissa täytyy olla yhtä paljon alkioita, tai jälkimmäinen saa olla vakio!')
26     disp('Siis kulma=[10,20,30];ja korkeus=[1,2,3] tai korkeus=20.')
27 end
28
29 for k=1:length(kulma)
30     view(kulma(k),korkeus(k))
31     title(['tarkastelusuunta: [az,el]=[', num2str(kulma(k)),',', num2str(korkeus(k)), ' ]'])
32     f(k)=getframe;
33 end
34
35
36 end
```


Tallennetaan syntynyt video

Listing 15: Tallennetaan syntynyt video

```
1 function [g] =videotallennus(f)
2 %Tallentaa videon, jossa voi olla eri kokoisia ruutuja.
3 %
4 %Esimerkiksi
5 %
6
7
8 %Etsitään ruutujen maksimikorkeus ja -leveys
9 [n,m]=size(f);
10 for k=1:m
11 [kor,lev,~]=size(f(k).cdata);
12 kork(k)=kor;leve(k)=lev;
13 end
14 mkor=max(kork);mlev=max(leve);
15
16 %Tehdään kaikista ruuduista samankokoisia
17 for k=1:m
18 g(k).cdata=uint8(240*ones(mkor,mlev,3));
19 g(k).cdata(1:kork(k),1:leve(k),:)=f(k).cdata;
20 g(k).colormap=[];
21 end
```

Listing 16: Tallennetaan syntynyt video

```
23 %Videon nimeksi voidaan asettaa kellonaika
24 vtyyppi='.avi'; %videon tiedostopaate
25 vnimi=['video',num2str(-floor(clock)),vtyyppi]; %poimitaan kellonaika
26 vnimi=vnimi(vnimi~=' '); %poistetaan valilyonnit
27
28 %v = VideoWriter(vnimi,'Uncompressed AVI'); %suuri ja tarkka video
29 v = VideoWriter(vnimi);
30 open(v)
31 writeVideo(v,g)
32 close(v)
33 end
```

4 Esimerkki: taso ja pyöritysvideon tekeminen siitä

Halutaan piirtää taso ja tehdä siitä pyöritysvideo. Valitaan tason pisteiksi $p = (1, 0, 0)$, $q = (0, 1, 0)$ ja $r = (0, 0, -0.3)$.

Ajetaan siis seuraava koodi, jonka voit halutessasi copypastettaa MATLABiin

```
%taso, yksikkönormaali, yksikkösuuntavektorit
p=[1 0 0];q=[0,1,0];r=[0,0,-0.3]; %määritellään pisteet
[a,b,c,d]=tasopqr2abcd(p,q,r); %tason yhtälön parametrit
t=linspace(-2,2,100); %olkoot x,y välillä [-2,2]
tasoabcd2kuva(a,b,c,d,t); %piirretään taso
[olahin] =tasonorigoalahin(a,b,c,d);%piirretään origoa lähin piste
tasokuvauvn(a,b,c,d) %piirretään suuntavektorit ja normaali

%pyöritetään ja tallennetaan video
kulma=linspace(1,360,360); %määritellään pyörityskulma
korkeus=20+10*sin(kulma/180); %määritellään vaihteleva korkeus
f=videopyor(kulma,korkeus); %tehdään pyöritysvideo
g=videotallennus(f); %tallennetaan video
```

Katso video osoitteesta <https://integraali.com/vektorilaskenta/matlab/tasopyoritysvideo.avi>

tai osoitteesta <https://www.youtube.com/watch?v=XWF10xZqoiw>