

# Mer om geometriska transformationer

## 1 Inledning

Vi fortsätter med geometriska transformationer och ser på ortogonal (vinkelrät) projektion samt spegling. Avslutningsvis skall vi även se på rotation runt sned axel i  $\mathbb{R}^3$ , förra laborationen roterade vi bara runt koordinataxlarna.

Men allra först ser vi kort hur man beräknar skalärprodukt, norm och liknande i MATLAB.

## 2 Skalärprodukt och norm i MATLAB

Skalärprodukten mellan två vektorer  $\mathbf{u}$  och  $\mathbf{v}$  ges av

$$\mathbf{u} \cdot \mathbf{v} = \mathbf{u}^T \mathbf{v} = u_1 v_1 + u_2 v_2 + u_3 v_3$$

och normen, som motsvarar absolutbelopp, ges av

$$\|\mathbf{u}\| = \sqrt{u_1^2 + u_2^2 + u_3^2} \quad (\|\mathbf{u}\|^2 = \mathbf{u} \cdot \mathbf{u})$$

Med MATLAB beräknar vi skalärprodukt och norm med funktionerna `dot` och `norm` enligt

```
>> dot(u,v)
>> norm(u)
```

Vinkeln  $\phi$  mellan två vektorer  $\mathbf{u}$  och  $\mathbf{v}$  ges av

$$\phi = \arccos \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\| \|\mathbf{v}\|}$$

och beräknas i MATLAB med

```
>> phi=acos(dot(u,v)/(norm(u)*norm(v)))
```

Vi påminner oss om att vektorerna  $\mathbf{u}$  och  $\mathbf{v}$  är ortogonala eller vinkelräta mot varandra om  $\mathbf{u} \cdot \mathbf{v} = 0$ , vilket ofta betecknas  $\mathbf{u} \perp \mathbf{v}$ .

En enhetsvektor är en vektor  $\mathbf{u}$  med  $\|\mathbf{u}\| = 1$ . Vill vi bestämma en enhetsvektor  $\mathbf{u}$  med samma riktning som vektorn  $\mathbf{v}$  så ges den av  $\mathbf{u} = \frac{1}{\|\mathbf{v}\|} \mathbf{v}$  och i MATLAB skulle vi skriva

```
>> u=v/norm(v)
```

Avståndet mellan två vektorer  $\mathbf{u}$  och  $\mathbf{v}$  ges av

$$\|\mathbf{u} - \mathbf{v}\| = \sqrt{(u_1 - v_1)^2 + (u_2 - v_2)^2 + (u_3 - v_3)^2}$$

och beräknas med

```
>> norm(u-v)
```

Dessa beräkningar görs på samma sätt för vektorer i  $\mathbb{R}^n$ , oavsett om  $n = 2, 3$ , eller större.

Även om vi inte använder den nu, så nämner vi kryssprodukten i  $\mathbb{R}^3$  som ges av

$$\mathbf{u} \times \mathbf{v} = (u_2v_3 - u_3v_2, u_3v_1 - u_1v_3, u_1v_2 - u_2v_1)$$

vilket är en vektor i  $\mathbb{R}^3$  och beräknas med funktionen `cross` enligt

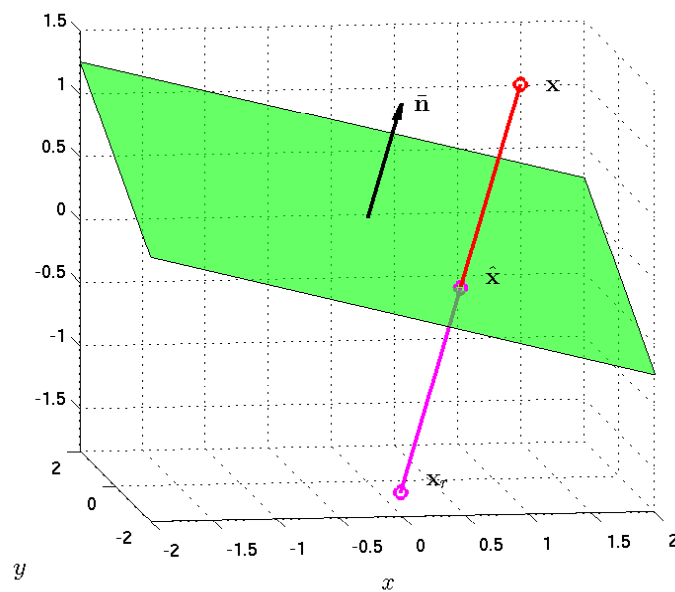
```
>> cross(u,v)
```

### 3 Ortogonal projektion och spegling

Vi skall bestämma ortogonala eller vinkelräta projektionen på planet

$$ax + by + cz = d$$

Planet har en normalvektor  $\mathbf{n} = (a, b, c)$ . I bilden har vi ritat en enhetsnormal  $\bar{\mathbf{n}}$  och projektionen  $\hat{\mathbf{x}}$  längs normalen av en punkt  $\mathbf{x}$ , dvs. den vinkelräta projektionen på planet.



Vi gör ansatsen  $\hat{\mathbf{x}} = \mathbf{x} + \alpha \mathbf{n}$  där  $\alpha$  skall bestämmas så att  $\hat{\mathbf{x}}$  ligger på planet.

Ekvationen för planet kan skrivas

$$\mathbf{n} \cdot \hat{\mathbf{x}} = d$$

och sätter vi in ansatsen får vi

$$\mathbf{n} \cdot \hat{\mathbf{x}} = \mathbf{n} \cdot (\mathbf{x} + \alpha \mathbf{n}) = \mathbf{n} \cdot \mathbf{x} + \alpha \mathbf{n} \cdot \mathbf{n} = d$$

och därmed

$$\alpha = \frac{d - \mathbf{n} \cdot \mathbf{x}}{\mathbf{n} \cdot \mathbf{n}}$$

För speglingen av  $\mathbf{x}_r$  av punkten  $\mathbf{x}$  i planet gäller

$$\mathbf{x}_r = \mathbf{x} + 2\alpha \mathbf{n}$$

med samma val av  $\alpha$ .

Nu skall vi i MATLAB rita planet  $ax + by + cz = d$ , för  $a = 1, b = -1, c = 4$  och  $d = 1$ . Eftersom  $c \neq 0$  så kan vi lösa ut  $z$ , i annat fall får vi modifiera koden

```

xmin=-2; xmax=2; ymin=-2; ymax=2;
a=1; b=-1; c=4; d=1;
X=[xmin xmax xmin xmin]; Y=[ymin ymin ymax ymax];
Z=(d-a*X-b*Y)/c;
fill3(X,Y,Z,'g','facealpha',0.7)
xlabel('x'), ylabel('y')
axis equal, grid on

```

Resultatet blir planet i figuren ovan. Egentligen är det ju bara en bit av planet, nämligen den del som ligger ovanför området  $-2 \leq x \leq 2$ ,  $-2 \leq y \leq 2$ .

**Uppgift 1.** Rita planet vi just tittade på. Bestäm en normalvektor och rita ut den med en pil från en punkt på planet. Pilar ritade vi i planet med `quiver` i höstas i samband med differential-ekvationer. När vi nu skall rita en pil i rummet kan vi göra det med `quiver3(x,y,z,a,b,c,s)`, där  $x, y, z$  ger koordinaterna för den punkt som pilen skall ritas från,  $a, b, c$  ger pilens utsträckning och  $s$  är en skalfaktor (normalt tar vi  $s = 0$  vilket ritas utan skalning, medan t.ex.  $s = 2$  gör pilarna dubbelt så långa). Välj en punkt  $\mathbf{x}$ , rita ut den, bestäm dess vinkelräta projektion  $\hat{\mathbf{x}}$  på planet och rita ut även den. Slutligen rita också ut speglingen  $\mathbf{x}_r$  av  $\mathbf{x}$ . Markera normalvektorn och de olika punkterna med texter. T.ex. normalvektorn kan markeras med `text(u,v,w,'n')`, där  $u, v, w$  är koordinaterna för positionen av vänster sida av texten och `'n'` är texten som skall skrivas, dvs. ett  $\mathbf{n}$ .

Snyggare blir det om vi sätter ut  $\bar{\mathbf{n}}$  med L<sup>A</sup>T<sub>E</sub>X enligt

```
text(u,v,w,'$\bar{\mathbf{n}}$', 'interpreter','latex','fontsize',12)
```

och på motsvarande sätt får vi  $\mathbf{x}$ ,  $\hat{\mathbf{x}}$  och  $\mathbf{x}_r$  med  `$\mathbf{x}$` ,  `$\hat{\mathbf{x}}$`  respektive  `$\mathbf{x}_r$` .

**Uppgift 2.** Rita samma plan i en ny bild. I samma bild skall ni rita en tetraeder. Tetraedern skall vara så placerad att ingen av dess sidor skär planet. Rita därefter projektionen och speglingen av tetraedern i planet.

## 4 Rotation runt sned axel i $\mathbb{R}^3$

Betrakta en punkt  $\mathbf{x} = (x_1, x_2, x_3)$  i  $\mathbb{R}^3$ . Antag vi vill rotera punkten runt en axel, vars riktning ges av en vektor  $\mathbf{v}$ , med en vinkel  $\phi$ . Rotationen skall göras medurs relativt riktningen på vektorn.

Vi kan då via ett basbyte återföra denna rotation till en rotation runt  $x_1$ -,  $x_2$ - eller  $x_3$ -axeln. För dessa har vi redan sett på standardmatriserna i förra laborationen.

Säg att vi vill återföra till en rotation runt  $x_1$ -axeln. Vi normaliserar  $\mathbf{v}$ , dvs. vi bildar  $\mathbf{v}_1 = \alpha \mathbf{v}$  där skalfaktorn  $\alpha$  väljs så att  $\mathbf{v}_1$  får enhetslängd. Därefter väljer vi två vektorer  $\mathbf{v}_2$  och  $\mathbf{v}_3$ , båda av enhetslängd, så att  $\mathbf{v}_2$  och  $\mathbf{v}_3$  är vinkelräta mot  $\mathbf{v}_1$  och vinkelräta mot varandra.

Vi skall helt enkelt se till att  $\{\mathbf{v}_2, \mathbf{v}_3\}$  blir en ortogonalbas för nollrummet  $\text{Nul}(\mathbf{v}_1^T)$ . Denna bas kan vi lätt beräkna i MATLAB med funktionen `null`.

Vi undersöker om  $\{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\}$  är en höger orienterad bas genom att beräkna determinanten

$$D = \det([\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3]).$$

Om  $D > 0$  så väljer vi  $\mathbf{b}_1 = \mathbf{v}_1$ ,  $\mathbf{b}_2 = \mathbf{v}_2$ ,  $\mathbf{b}_3 = \mathbf{v}_3$  annars tar vi  $\mathbf{b}_1 = \mathbf{v}_1$ ,  $\mathbf{b}_2 = \mathbf{v}_3$ ,  $\mathbf{b}_3 = \mathbf{v}_2$ .

Nu bildar vi basbytesmatrisen  $\mathbf{P} = [\mathbf{b}_1 \ \mathbf{b}_2 \ \mathbf{b}_3]$  och eftersom kolonnerna ortogonala och normerade så gäller  $\mathbf{P}^{-1} = \mathbf{P}^\top$ .

Standardmatrisen för rotation runt axeln som ges av  $\mathbf{v}$  blir

$$\mathbf{A}_v = \mathbf{P} \mathbf{A} \mathbf{P}^{-1} = \mathbf{P} \mathbf{A} \mathbf{P}^\top$$

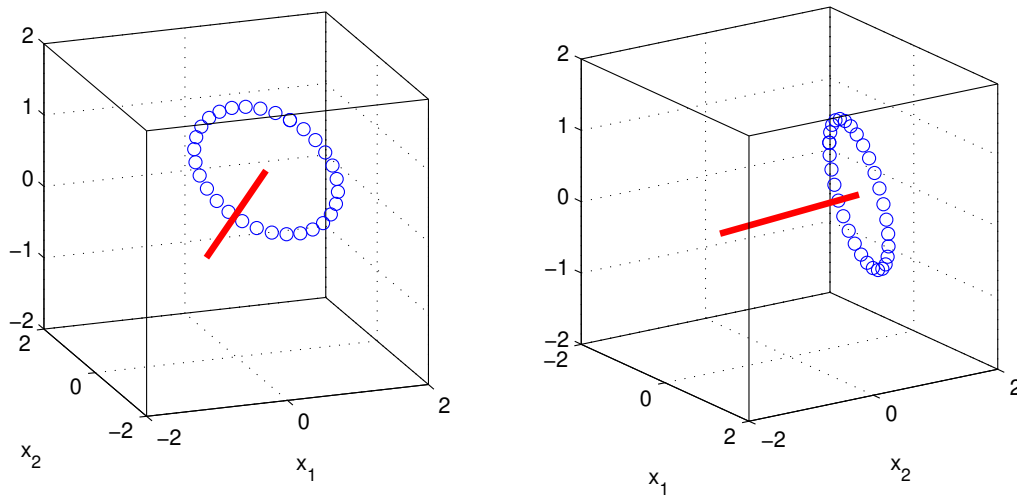
där

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) & \cos(\phi) \end{bmatrix}$$

är standardmatrisen för rotationen runt  $x_1$ -axeln.

Detaljerna ovan tar nog lite tid att förstå, vi utnyttjar både vektorrum och basbyte. Det hindrar inte att vi kan rotera i MATLAB redan nu.

Nedan ser vi en rotation av en punkt runt en viss axel, upprepad några gånger, sett från två olika betraktelsevinklar.



Så här gjorde vi en skriptfil i MATLAB

```
phi=pi/15;
A=[1 0 0; 0 cos(phi) -sin(phi); 0 sin(phi) cos(phi)];
v=[2;2;1]; v=v/norm(v); Z=null(v'); P=[v,Z];
if det(P)<0, P(:,[2 3])=P(:,[3 2]); end
Av=P*A*P';
x=[0.8; 0.1; 1.2];
plot3(x(1),x(2),x(3),'o'), hold on
for i=1:30
    x=Av*x;
    plot3(x(1),x(2),x(3),'o')
end
plot3([-v(1) v(1)],[-v(2) v(2)],[-v(3) v(3)],'r','linewidth',3)
box on, grid on, hold off
axis equal, axis([-2 2 -2 2 -2 2]), axis vis3d
```

**Uppgift 3.** Roterar, med en liten vinkel upprepade gånger, en punkt runt någon sned axel som ni själva väljer (dock inte samma punkt och axel som i exemplet).

**Uppgift 4.** *Frivillig!* Roterar en tetraeder runt någon sned axel som ni själva väljer. Det skall vara en animering, precis som i uppgift 4 i förra laborationen. Se till att tetraederna är placerade i förhållande till axeln så att rotationen syns tydligt. Tetraedern får givetvis inte deformeras under rotationen.

## 1 Målsättning

Denna laborationen är en fortsättning på den förra. Vi utökar transformationerna med projektion och spegling samt rotation runt godtycklig axel.

## 2 Kommentarer och förklaringar

Vi skall se lite mer på den ortogonala projektionen på ett plan  $ax + by + cz = d$  där  $a, b, c$  och  $d$  är konstanter.

Om  $\mathbf{x}$  är den punkt vi projicerar och  $\hat{\mathbf{x}}$  är projektionen så gäller

$$\hat{\mathbf{x}} = \mathbf{x} + \alpha \mathbf{n}, \quad \alpha = \frac{d - \mathbf{n} \cdot \mathbf{x}}{\mathbf{n} \cdot \mathbf{n}}$$

där  $\mathbf{n} = (a, b, c)$  är en normalvektor till planet.

Om  $d = 0$ , dvs. planet går genom origo, har vi

$$\hat{\mathbf{x}} = \mathbf{x} + \alpha \mathbf{n} = \mathbf{x} - \frac{\mathbf{n} \cdot \mathbf{x}}{\mathbf{n} \cdot \mathbf{n}} \mathbf{n}$$

Eftersom  $\mathbf{n} \cdot \mathbf{x}$  är en skalär så gäller

$$\hat{\mathbf{x}} = \mathbf{x} - \frac{(\mathbf{n} \cdot \mathbf{x})}{\mathbf{n} \cdot \mathbf{n}} \mathbf{n} = \mathbf{x} - \frac{1}{\mathbf{n} \cdot \mathbf{n}} \mathbf{n} (\mathbf{n} \cdot \mathbf{x})$$

Vidare gäller att  $\mathbf{n} \cdot \mathbf{x} = \mathbf{n}^T \mathbf{x}$ , dvs. skalärprodukten kan beräknas genom att  $\mathbf{n}^T$  som är en radvektor matrismultipliceras med kolonnvektorn  $\mathbf{x}$  och vi får

$$\hat{\mathbf{x}} = \mathbf{x} - \frac{1}{\mathbf{n} \cdot \mathbf{n}} \mathbf{n} (\mathbf{n}^T \mathbf{x}) = \mathbf{x} - \frac{1}{\mathbf{n} \cdot \mathbf{n}} (\mathbf{n} \mathbf{n}^T) \mathbf{x} = \left( \mathbf{I} - \frac{1}{\mathbf{n} \cdot \mathbf{n}} \mathbf{n} \mathbf{n}^T \right) \mathbf{x}$$

där vi utnyttjade att  $\mathbf{n} (\mathbf{n}^T \mathbf{x}) = (\mathbf{n} \mathbf{n}^T) \mathbf{x}$ .

Observera att  $\mathbf{n} \mathbf{n}^T$  är en matris (en s.k. ytterprodukt), medan  $\mathbf{n} \cdot \mathbf{n} = \mathbf{n}^T \mathbf{n}$  är ett tal (skalärprodukt eller innerprodukt).

Vi har kommit fram till

$$\hat{\mathbf{x}} = \left( \mathbf{I} - \frac{1}{\mathbf{n} \cdot \mathbf{n}} \mathbf{n} \mathbf{n}^T \right) \mathbf{x} = \mathbf{P} \mathbf{x}$$

Alltså en linjär avbildning med standardmatrisen  $\mathbf{P}$ .

För spegling gäller

$$\mathbf{x}_r = \left( \mathbf{I} - \frac{2}{\mathbf{n} \cdot \mathbf{n}} \mathbf{n} \mathbf{n}^T \right) \mathbf{x} = \mathbf{R} \mathbf{x}$$

Alltså en linjär avbildning med standardmatrisen  $\mathbf{R}$ .

Om  $d \neq 0$ , dvs. planet går *inte* genom origo, har vi

$$\hat{\mathbf{x}} = \mathbf{x} + \alpha \mathbf{n} = \mathbf{x} + \frac{d - \mathbf{n} \cdot \mathbf{x}}{\mathbf{n} \cdot \mathbf{n}} \mathbf{n} = \left( \mathbf{I} - \frac{1}{\mathbf{n} \cdot \mathbf{n}} \mathbf{n} \mathbf{n}^\top \right) \mathbf{x} + \frac{d}{\mathbf{n} \cdot \mathbf{n}} \mathbf{n} = \mathbf{P} \mathbf{x} + \beta \mathbf{n}$$

Detta är en affin avbildning som vi beskriver med (homogena koordinater)

$$\begin{bmatrix} \hat{\mathbf{x}} \\ \hat{w} \end{bmatrix} = \begin{bmatrix} \mathbf{P} & \beta \mathbf{n} \\ \mathbf{0}^\top & 1 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ w \end{bmatrix}$$

dvs. med hjälp av matrismultiplikation.

Motsvarande gäller för spegling, då  $d \neq 0$ .

### 3 Lärandemål

Efter denna laboration skall du

- kunna projicera och spegla ett objekt i ett plan
- kunna rotera ett objekt runt en godtycklig axel