

Uppgift 1. Bestäm alla lösningar till ekvationssystemet

$$\begin{cases} 2 \sin(x_1 x_2) - x_1 + 1.5 = 0 \\ \cos(x_1 - 3x_2) - 2x_2 + 1 = 0 \end{cases}$$

(a). Inför en funktion $\mathbf{f}: \mathbb{R}^2 \rightarrow \mathbb{R}^2$ så att vi kan skriva ekvationerna på den kompakta formen

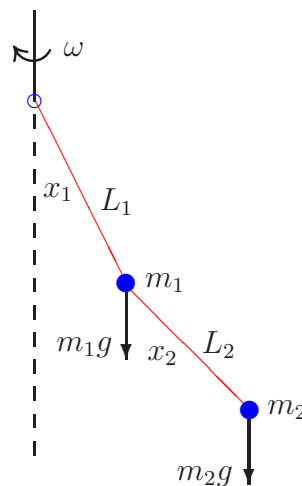
$$\mathbf{f}(\mathbf{x}) = \mathbf{0}$$

(b). Beskriv i MATLAB funktionen $\mathbf{f}(\mathbf{x})$ på ett sådant sätt så att du lätt kan rita nollnivåkurvor. Skriv även ned den kod som behövs för att rita nollnivåerna för f_1 och f_2 .

(c). Beräkna derivatamatrixen $D\mathbf{f}(\mathbf{x})$ och beskriv den i MATLAB.

(d). Skriv en kod i MATLAB som läser in en startapproximation \mathbf{x}_0 och löser ekvationen $\mathbf{f}(\mathbf{x}) = \mathbf{0}$ med Newtons metod.

Uppgift 2. En dubbel pendel roterar med vinkelhastigheten ω runt en vertikal axel.



Vid jämvikt för de två pendlarna är vinklarna mot vertikala axeln, x_1 och x_2 , lösningar till följande system av ekvationer

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} f_1(x_1, x_2) \\ f_2(x_1, x_2) \end{bmatrix} = \begin{bmatrix} \tan(x_1) - \frac{\omega^2}{g} (L_1 \sin(x_1) + \frac{m_2}{m_1+m_2} L_2 \sin(x_2)) \\ \tan(x_2) - \frac{\omega^2}{g} (L_1 \sin(x_1) + L_2 \sin(x_2)) \end{bmatrix} = \mathbf{0}$$

Om vi tar $m_1 = m_2$ och $L_1 = L_2 = L$ så får vi funktionen

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} \tan(x_1) - k(\sin(x_1) + \frac{1}{2} \sin(x_2)) \\ \tan(x_2) - k(\sin(x_1) + \sin(x_2)) \end{bmatrix}$$

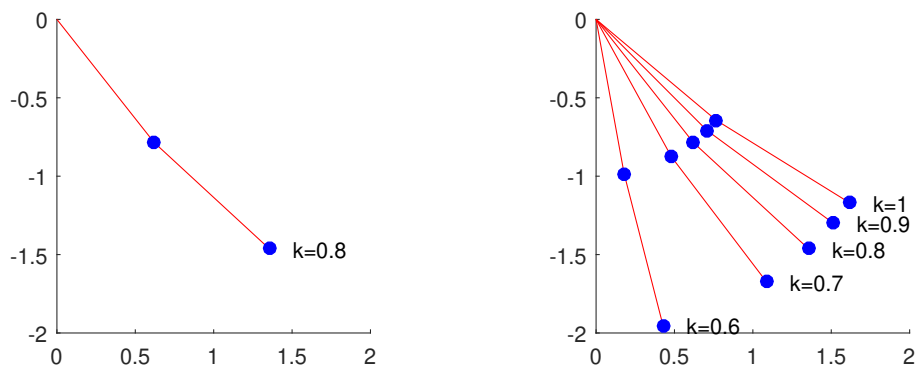
med $k = \frac{\omega^2 L}{g}$ som parameter.

(a). Beskriv i MATLAB funktionen $f(\mathbf{x})$ på ett sådant sätt så att du lätt kan rita nollnivåkurvor. Skriv även ned den kod som behövs för att rita nollnivåerna för f_1 och f_2 . Vilket område i planet ritas du över.

(b). Beräkna derivatamatrixen $Df(\mathbf{x})$ och beskriv den i MATLAB.

(c). Skriv en kod i MATLAB som löser ekvationen $f(\mathbf{x}) = \mathbf{0}$ med Newtons metod. Tag $k = 0.8$ och antag att vi har en startapproximation av en lösning \mathbf{x}_0 .

(d). Här har vi ritat upp pendlarna, till vänster för $k = 0.8$ och till höger för några olika k -värden. Vid lösning av ekvationen har värdet på L varit inbakat i k -värdena, men vid uppritningen har vi tagit $L = 1$.



Skriv en kod i MATLAB för att rita figuren ovan till vänster.

Uppgift 3. Vi skall beräkna alla lokala extremvärden till

$$f(\mathbf{x}) = x_1^4 + 3x_2^4 + x_1x_2 - x_1^2 - x_2^2$$

(a). Beskriv i MATLAB funktionen $f(\mathbf{x})$ på ett sådant sätt så att du lätt kan rita nivåkurvor. Skriv även ned den kod som behövs för att rita nivåkurvorna. Se till att det blir lätt att ändra vilket område i planet du ritas över.

(b). Beräkna gradienten $\nabla f(\mathbf{x})$ och Hessematrixen $\mathbf{H}(\mathbf{x})$ och beskriv dem i MATLAB. Skriv en kod som ritas noll-nivåkurvor till de två komponenterna i gradientvektorn.

(c). Beräkna alla stationära punkter noggrant. Skriv en kod i MATLAB som löser ekvationen $\nabla f(\mathbf{x}) = \mathbf{0}$ med Newtons metod. Beskriv även hur vi enkelt kan vi läsa av startapproximationer. Skriv en kod som avgör vilken typ av stationära punkter vi har med hjälp av egenvärdena till Hessematrixen $\mathbf{H}(\mathbf{x})$.

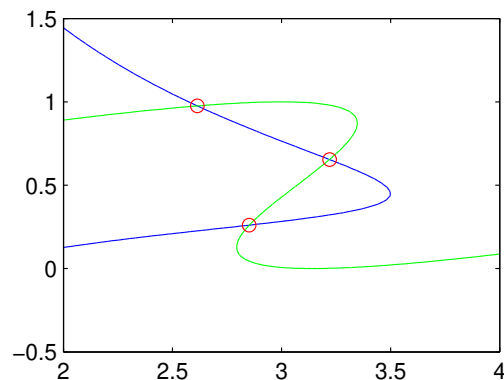
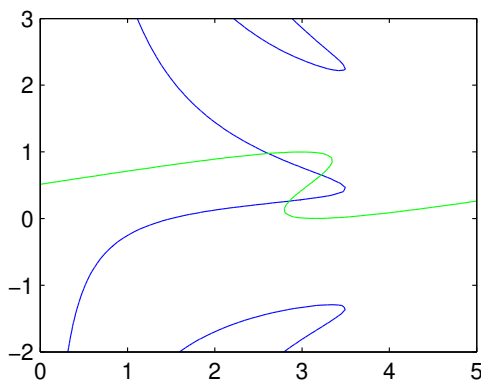
Uppgift 1(a).

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} 2 \sin(x_1 x_2) - x_1 + 1.5 \\ \cos(x_1 - 3x_2) - 2x_2 + 1 \end{bmatrix}$$

(b).

```
f1=@(x1,x2)2*sin(x1.*x2)-x1+1.5;
f2=@(x1,x2)cos(x1-3*x2)-2*x2+1;
%%

x1=linspace(0,5,80); x2=linspace(-2,3,80);
[X1,X2]=meshgrid(x1,x2);
F1=f1(X1,X2); F2=f2(X1,X2);
clf
contour(x1,x2,F1,[0 0], 'b')
hold on
contour(x1,x2,F2,[0 0], 'g')
```



Vi ritar först bilden till vänster, sedan den till höger där vi tagit ett mindre område i planet. I höger bild ser vi även de lösningar vi ritat ut med koden för Newtons metod i (d).

(c).

$$D\mathbf{f}(\mathbf{x}) = \begin{bmatrix} 2x_2 \cos(x_1 x_2) - 1 & 2x_1 \cos(x_1 x_2) \\ -\sin(x_1 - 3x_2) & 3 \sin(x_1 - 3x_2) - 2 \end{bmatrix}$$

```
Df=@(x) [2*x(2)*cos(x(1)*x(2))-1 2*x(1)*cos(x(1)*x(2))
         -sin(x(1)-3*x(2))      3*sin(x(1)-3*x(2))-2 ];
```

(d).

```
f=@(x) [f1(x(1),x(2))
        f2(x(1),x(2))];
```

Vi läser in startapproximation och löser upprepade gånger med

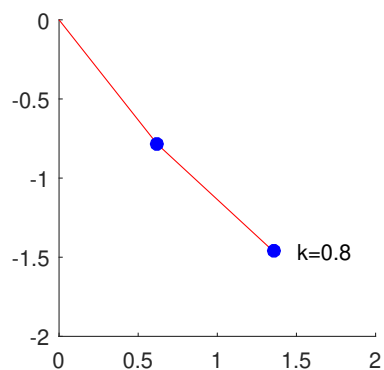
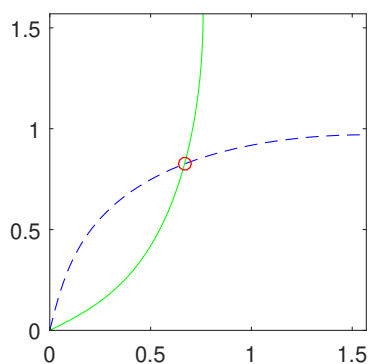
```
[x10,x20]=ginput(1); x=[x10;x20];
kmax=10; tol=0.5e-5;
for k=1:kmax
    d=-Df(x)\f(x);
    x=x+d;
    if norm(d)<tol, break, end
end
disp([x' norm(d)])
plot(x(1),x(2),'ro')

% Detta är utskriften
% 2.8516 0.26022 5.1326e-009
% 3.2199 0.6542 3.1859e-010
% 2.6142 0.97571 8.6706e-011
```

Uppgift 2(a).

```
kp=0.8; % Vi använder variabeln kp för parametervärdet
        % eftersom k används i Newtons metod.
f1=@(x1,x2)tan(x1)-kp*(sin(x1)+0.5*sin(x2));
f2=@(x1,x2)tan(x2)-kp*(sin(x1)+sin(x2));
%%

x1=linspace(0,pi/2,80); x2=linspace(0,pi/2,80);
[X1,X2]=meshgrid(x1,x2);
F1=f1(X1,X2); F2=f2(X1,X2);
clf
subplot(1,2,1)
contour(x1,x2,F1,[0 0],'b')
hold on
contour(x1,x2,F2,[0 0],'g')
axis equal
```



Till vänster syns nivåkurvorna, där vi även ser lösningen markerad av koden i (c). Pendeln uppritad i (d) ser vi till höger.

(b).

$$Df(x) = \begin{bmatrix} 1 + \tan^2(x_1) - k \cos(x_1) & -\frac{1}{2}k \cos(x_2) \\ -k \cos(x_1) & 1 + \tan^2(x_2) - k \cos(x_2) \end{bmatrix}$$

```
Df=@(x) [1+tan(x(1))^2-kp*cos(x(1)) -0.5*kp*cos(x(2))
        -kp*cos(x(1))                1+tan(x(2))^2-kp*cos(x(2))];
```

(c).

```
f=@(x) [f1(x(1),x(2))
        f2(x(1),x(2))];

[x10,x20]=ginput(1); x=[x10;x20];
kmax=10; tol=0.5e-5;
for k=1:kmax
    d=-Df(x)\f(x);
    x=x+d;
    if norm(d)<tol, break, end
end
disp([x' norm(d)])
plot(x(1),x(2),'ro')

% Utskriften
% 0.66851 0.8256 1.6579e-007
```

(d).

```
subplot(1,2,2)
Px=[0 sin(x(1)) sin(x(1))+sin(x(2))];
Py=[0 -cos(x(1)) -cos(x(1))-cos(x(2))];
plot(Px,Py), hold on
text(Px(3)+0.15,Py(3),['k=',num2str(kp)])
plot(Px(2:3),Py(2:3),'o')
axis equal, axis([0 2 -2 0])
```

Uppgift 3(a).

```
f=@(x1,x2)x1.^4+3*x2.^4+x1.*x2-x1.^2-x2.^2;
%%

x1=linspace(-1.5,1.5,80); x2=linspace(-1.5,1.5,80);
[X1,X2]=meshgrid(x1,x2);
F=f(X1,X2);
clf
subplot(2,2,1)
surf(x1,x2,F)
axis tight, xlabel('x_1'), ylabel('x_2')
subplot(2,2,2)
contour(x1,x2,F,linspace(-1,1,80))
xlabel('x_1'), ylabel('x_2')
```

(b). Gradienten $\nabla f(\mathbf{x})$

```
df1=@(x1,x2)4*x1.^3+x2-2*x1;  
df2=@(x1,x2)12*x2.^3+x1-2*x2;
```

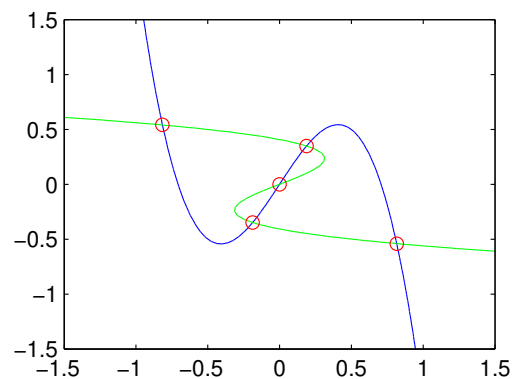
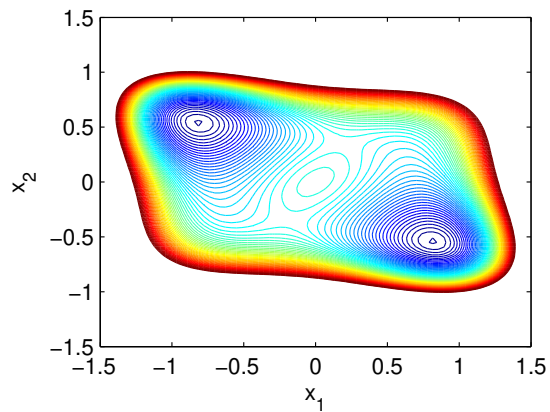
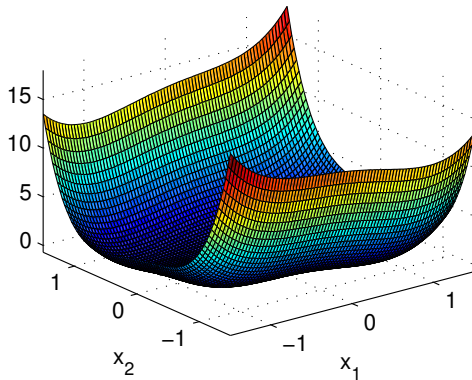
```
gradf=@(x) [df1(x(1),x(2))  
            df2(x(1),x(2))];
```

Hessematrisen $\mathbf{H}(\mathbf{x})$

```
H=@(x) [12*x(1)^2-2 1  
        1 36*x(2)^2-2];
```

Noll-nivåkurvor

```
subplot(2,2,4)  
contour(x1,x2,df1(X1,X2), [0 0], 'b')  
hold on  
contour(x1,x2,df2(X1,X2), [0 0], 'g')
```



(c). Vi gör följande upprepade gånger

```
[x10,x20]=ginput(1); x=[x10;x20];  
kmax=10; tol=0.5e-5;  
for k=1:kmax  
    d=-H(x)\gradf(x);  
    x=x+d;
```

```

    if norm(d)<tol, break, end
end
plot(x(1),x(2),'ro')
eg=eig(H(x));

disp(['x=( ',num2str(x(1)),' ',num2str(x(2)),'), |d|= ',num2str(norm(d))])
disp(['Egenvärden ',num2str(eg')])

% Utskriften
% x=( -0.8159, 0.54074), |d|= 8.9765e-008
% Egenvärden 5.6416      8.8732
%
% x=( 0.18777, 0.34905), |d|= 3.4936e-010
% Egenvärden -1.815     2.6242
%
% x=( 1.9722e-031, -3.9443e-031), |d|= 3.1415e-015
% Egenvärden -3 -1
%
% x=( -0.18777, -0.34905), |d|= 3.22e-008
% Egenvärden -1.815     2.6242
%
% x=( 0.8159, -0.54074), |d|= 7.2872e-011
% Egenvärden 5.6416     8.8732

```