1 Varmajafnvægi

1.1

Skrifa `matlab`-forrit sem finnur nálgunarlausn á Sturm-Liouville-verkefninu (1.1) með Galerkin-aðferð og þúfugrunnföllum, þar sem gert er ráð fyrir að p megi vera ósamfellt og að hægri hliðin megi innihalda punktuppsprettu.
Lausn: Forritið er svona útlítandi

```
%% FAllið Galerkinadferd er gert til þess að fá nálgun með aðferð Galerkins á þúfugrunnföllum.
% Inntök fyrir fallið eru föllin p(x),q(x),f(x) ásamt jaðargildisstuðlum: alpha, beta og gamma.
% Einnig býður fallið upp á að notandi setji inn punktuppsprettur í punkti r af stærð Q.
% Fallið skilar svo vigri c með stuðlum c_i sem er nálgun á lausn u(x_i).
% ATH: scriptið er að mestu skrifað eftir uppskrift frá Ragnari Sigurðssyni úr fyrirlestranótum hans frá janúar 2015.

function c=Galerkinadferd(p,q,f,x,alpha,beta,gamma,r,Q)

N=length(x); % Skilgreinum N sem er fjöldi punkta.
h=x(2:N)-x(1:N-1); % h er lengd hlutbils.
% Skilgreinum punkta m sem eru staðsetningar á millipunktum.
m=(x(2:N)+x(1:N-1))/2;
pm=p(m);
qm=q(m);
fx=f(x);
% Byrjum á að búa til núllfylki og núll-vigur fyrir A og b,
A=zeros(N,3);
b=zeros(N,1); % við fyllum svo jafnt og þétt inni í A og b.

% Byrjum á því að fylla inn í A og b fyrir línur 2 og niður í N-1:
A(2:N-1,2) = pm(1:N-2)./h(1:N-2)+pm(2:N-1)./h(2:N-1)+(h(1:N-2).*qm(1:N-2)+h(2:N-1).*qm(2:N-1))/3;
A(2:N,1) = -pm(1:N-1)./h(1:N-1)+h(1:N-1).*qm(1:N-1)/6;
A(1:N-1,3) = A(2:N,1);
b(2:N-1) = (h(1:N-2).*(fx(1:N-2)+2*fx(2:N-1))+h(2:N-1).*(2*fx(2:N-1) +fx(3:N)))/6;
% Fyllum svo inn í fyrstu línurnar, eftir því hvað betan er.
if beta(1) == 0
    A(1,2) = alpha(1);
    A(1,3) = 0;
    b(1) = gamma(1);
else
    A(1,2) = pm(1)/h(1)+h(1)*qm(1)/3+p(x(1))*alpha(1)/beta(1);
```

```
    b(1) = h(1)*(2*fx(1)+fx(2))/6+p(x(1))*gamma(1)/beta(1);
end

if beta(2) == 0
    A(N,1) = 0;
    A(N,2) = alpha(2);
    b(n) = gamma(2);
else
    A(N,2)= pm(N-1)/h(N-1)+h(N-1)*(qm(N-1)/3)+(p(x(N))*(alpha(2)/beta(2)));
    b(N) = h(N-1)*(fx(N-1)+2*fx(N))/6+p(x(N))*(gamma(2)/beta(2));
end
% Stingum svo inn punktuppsprettum ef þær eru til staðar, en ef engar uppsprettur eru kallaðar með fallinu, sleppir forritið þessum lið.
if ~isempty(r)
m=length(r);
for j=1:m

    i=max(find(x<=r(j)));
    b(i)= b(i)+Q(j);

end
end


%Notumst svo við forrit tridiagonal_solve sem Ragnar gaf á Uglu til að leysa út fyrir c.
c=tridiagonal_solve(A,b);
plot(x,c)
```

1.2

Rökstyðjið að aðferð Galerkins virki með ósamfellt p eins og lýst er hér að framan. Það eina sem þarf að gera að sýna að hlutheildunin sem liggur t il grundvallar aðferðinni virki í þessu tilfelli þegar p er ósamfellt eins og lýst er hér að framan.

Lausn:

Við getum sýnt fram á virkni Galerkins aðferðarinnar fyrir ósamfellt p með því að skipta bilinu sem heilda á yfir, upp í búta.

Ef við ætlum okkur að beita aðferðinni á bili [a,b] þar sem fall p er ósamfellt í endanlega mörgum innri-punktum s_i á bilinu, nægir okkur að beita aðferð Galarkins á búta

$$[a,s_i],\ [s_i,s_{i+1}],\cdots,\ [s_n,b].$$

Ljóst er að þegar lagaðar eru saman lausnir fyrir hvern bút, þá munu liðir háðir p(s_i) styttast út hver á móti örðum.

Við getum því notfært okkur þetta og lagt saman heildin fyrir hvert hlutbil til þess að fá nálgunina fyrir allt bilið.
Skoðum tilfellið þar sem p er ósamfellt í einum punkti s á bili [a,b]. Heildum þá yfir búta [a,s] og [s,b] og fáum:

$$\int_a^s -(p(x) * u(x)')dx + \int_a^s q(x)u(x)dx = \int_a^s f(x)$$
$$\text{og } \int_s^b -(p(x) * u(x)')dx + \int_s^b q(x)u(x)dx = \int_s^b f(x)$$

Ef við leggjum svo heildin saman fáum við:

$$(-p(s)u(s)' + p(a)u(a)') + (-p(b)u(b)' + p(s)u(s)') + \int_a^s q(x)u(x)dx + \int_s^b q(x)u(x)dx = \int_a^s f(x) + \int_s^b f(x)$$

Sem gefur svo:

$$\int_a^b -(p(x) * u(x)')dx + \int_a^b q(x)u(x)dx = \int_a^b f(x)$$

Það er því ljóst að punktur $p(s)$ hefur ekki áhrif á niðurstöðun og því virkar Galerkin aðferðin fyrir ósamfellt p. En til þess að svo megi vera þarf að gilda að þar sem $p(s_i)$ er ósamfellt, þá þarf $p(s_i) * u'(s_i)$ að vera samfellt og $u'(s_i)$ ósamfellt.

1.3

Gerið grein fyrir því hvernig punktuppspretturnar eru meðhöndlaðar í forritinu. Sýnið með einni keyrslu hvaða áhrif það hefur á lausn ef punktuppsprettu er bætt við hægri hliðina.
Lausn:
Við höfum gert ráð fyrir að hægri hliðin f í afleiðujöfnunni sé samfellt. Getum líka bætt við það liðum sem eru margfeldi fasta og Dirac-delta-falls. Slíkir liðir lýsa orkuuppsprettum í stökum punktum. Dirac-fallið í punktinum r táknum við með δ_r . Eina reiknireglan sem við þurfum að kunna er að

$$\int_c^d \delta_r(x)\varphi(x)dx = \varphi(r)$$

fyrir sérhvert fall φ á grennd um lokaða bilið $[c, d]$ og sérhvern punt $r \in [c, d]$. Af þessu leiðir að ef $r = x_k$ er einn af punktunum í skipitingu á bilinu [a,b] og φ_j er þúfugrunnfallið númer j, þá er

$$\int_\alpha^\beta \delta_{x_k}(x)\varphi_j(x)dx = \varphi_j(x_k) = \begin{Bmatrix} 1, j = k, \\ 0, j \neq k. \end{Bmatrix}$$

Í forritinu bætum við því punktuppsprettunum við hægri hlið vigursinns b, með eftirfarandi kóða:

```
%% Hér sést liðurinn úr fallinu Galerkinadferd sem bætir
    punktuppspretum við hægri hlið.

% Stingum svo inn punktuppsprettum ef þær eru til staðar, en ef
    engar uppsprettur eru kallaðar með fallinu, sleppir forritið
    þessum lið.
if ~isempty(r)
m=length(r);
for j=1:m

    i=max(find(x<=r(j)));
    b(i)= b(i)+Q(j);

end
end
```

1.4

Finnið sjálf heppilegt sýnidæmi til þess að prófa hvort forritið virki réttt. Kannið hvort skekkjan í nálguninni er $O(h^2)$, þar sem h táknar billengd í jafnri skiptingu með því að prófa forritið í sýnidæminu með fjölda bila N=2,4,8....

Lausn:

Heppilegt sýnidæmi sem við fundum er í heimadæmum 8.1. Byrjum á því að skilgreina breyturnar.

```
p = @(x) x;
q = @(x) 1./x;
f = @(x) -2 + 0.*x;
u=@(x) x.*log(x)-x;
x = [1:1/3:2];
alpha=[0,1];
beta = [-1,-2];
gamma=[0,-2];
r=2;
Q=0;
```

Köllum svo á fallið.

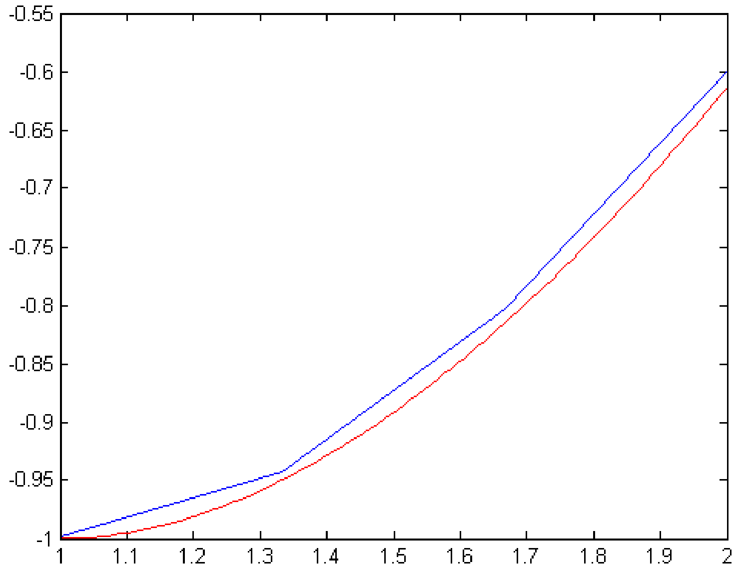
```
>> galerkinadferd_2(@(x)x,@(x)(1./x),@(x)-2+0.*x,[1:1/3:2],[0,1],[-1,-2],[0,-2],2,0)

ans =

   -0.9977
```

```
   -0.9424
   -0.8039
   -0.5987
```

Plottum svo upp.

```
>> plot(linspace(1,2,4),ans)
>> hold on
>> plot(linspace(1,2,100),u(linspace(1,2,100)),'r')
```

Fáum þá út fallið og nálgunarfallið.



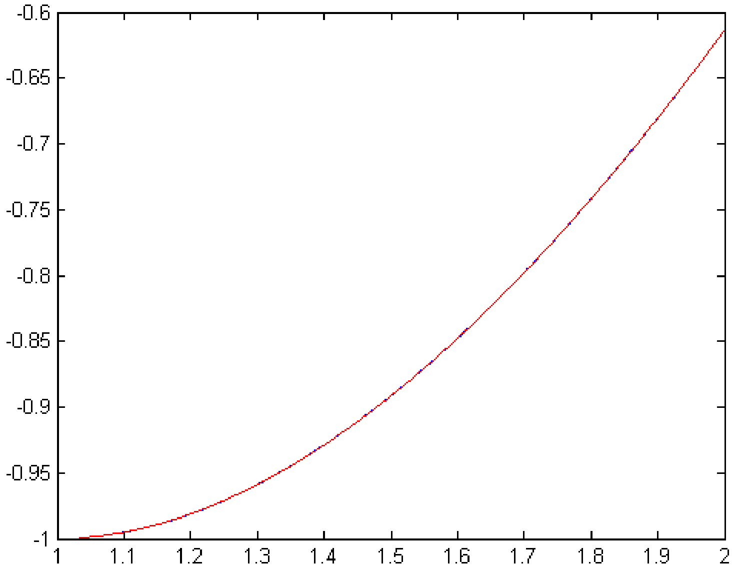
Mynd 1: Mynd af fallinu og nálguninni.

Sjáum hér að nálguninn er góð enn ef að við fjölgum bilunum þá verður hún ennþá meira nákvæm. Sjáum hér að neðan mynd þar sem bilin eru orðin 1000 og þá eru línurnar tvær svo gott sem samliggjandi.

2 Straujárnið hennar Mömmu

2.1

Skrifið `matlab`-forrit sem notar forritið ykkar`Galerkinadferd` úr síðasta lið til þess að finna nálgun á lausn þessa verkefnis. Notið fjögurra punkta skiptingu $x_0 = a, x_1 = s, x_2 = r$ og $x_3 = b$. Framkvæmið fastapunktsítrekunina þar til endurbítin í ítrekuninni er innan við 1%. Markmiðið er að þið



Mynd 2: Mynd af fallinu og nálguninni.

prófið ykkur áfram með að stilla gildið á Q þannig að út komi hitastigið $u(b) \approx 200C$. (Hámarksaflið í venulegu straujárni er 1200-1500W.) Þegar þetta gildi á Q er fundið eigði þi ða teikna upp hitastigið sem fall af x.

Lausn:

Byrjum á því að skilgreina allar breyturnar.

```
h1 = 5;
h2 = 50;
e = 0.2;
sigma = 5.669*10^-8;
T0 = 298;
T1 = 298;
l1 = 0.1;
l2 = 140;
gamma1 = h1*T0;
a = 0;
s = 0.008;
r = 0.01;
b = 0.016;
x =[0, 0.0008, 0.01, 0.016];
q = @(x) x.*0;
p = @(x) l1 + (l2 - l1)*(x >= r); %þetta er heavisidelidur
f = @(x) x.*0;
```

```
beta = [0.1,140]; %l1,l2
Q = 235/0.024;
Q_0 = Q/0.024;
v=[473];
skekkja = 1;
```

Nú keyrum við eftirfarandi forrit og fáum út hitastig.

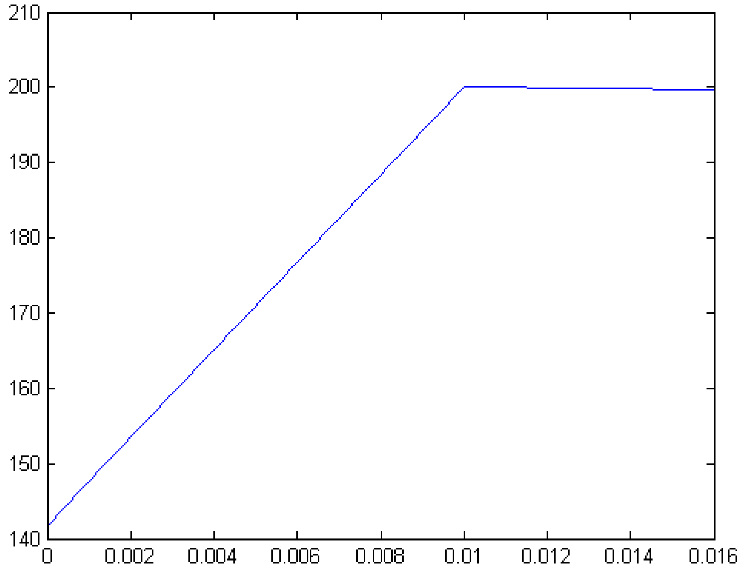
```
while skekkja>0.01
    %itra galerkin til að haekka hk
    hk=h2+e*sigma*(v(end) +T1)*(v(end)^2+T1^2);
    gamma=[h1*T0,hk*T1];
    alpha=[h1,hk];
    v1=v(end);
    v = galerkinadferd_2( p,q,f,x,alpha,beta,gamma,r,Q );
    skekkja=abs(v1-v(end))/v1;
end
disp(v(end))
plot(x,(v-273))

  472.6239
```

Sjáum að hitastigið sem kemur út er í kelvingráðunm og því drögum við frá 273 til að fá út hitastigið í celsíus $472.6239 - 273 = 199.6239$ og við tökum okkur það bessaleyfi að námunda það upp í 200°C og svo plottum við einnig mynd þar sem við sjáum hitastigið á grafi.
Sjáum á ferlinu að hitastigið hækkar hratt upp og svo helst það í sirka 200°C.

2.2

Leysið verkefnið aftur með gildið á Q sem þið fundið, en sleppið geislunarliðnum. ÞEtta þýðir að þið eigið að setja $\epsilon = 0$. Hver er munurinn á lausnunum? Er nauðsynlegt að taka tillit til geislunarliðarins.
Lausn:
við keyrum nákvæmlega sama í gegn í Matlab nema við breytum einni breytu þannig að `e=0` þá sjáum við að myndin breytist og hitastigið skekkist um sirka 8°C

```
481.5692

>> 481.5692-273

ans =
```



Mynd 3: Hitastigsmynd af straujárninu hennar Mömmu.

```
  208.5692
```

Nú skulum við skoða myndina. SJáum að hún er alveg eins nema bara með öðruvísi hitastig.

2.3

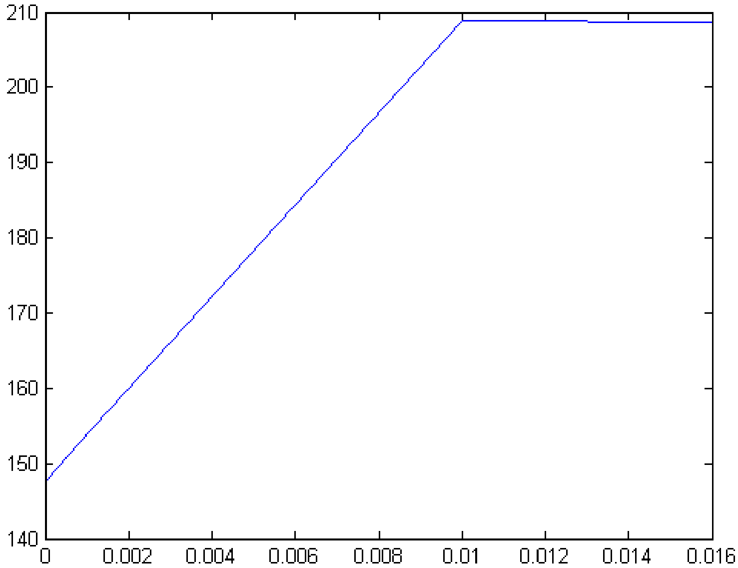
Útskýrið hvað gerist ef tekin er fínni skiptinn
Lausn:
Ef tekin er fínni skipting þá koma gildi sem eru einfaldlega nær 200°C.

3 Dirichlet verkefnið á rétthyrningi

3.1

Útfæra Dirichletverkfni í `matlab`.
Lausn:

```
function [W,A,RHS]=Dirichletverkefni(a,b,c,d,p,q,f,gamma,P,Q,R,U,h)
%Author: Andri Freyr Thorgeirsson, Erna Gudrun Thorsteinsdottir og
%        Rikhardur Thor Rognvaldsson
```



Mynd 4: Hitastigsmynd af straujárninu hennar Mömmu.

```
% Breytur

% Setja upp reiknineetid
M=(b-a)/h+1;                 % Fjoldi hnitpunkta i x att
N=(d-c)/h+1;              % Fjoldi hnitpunkta i y att
% Skilgreinum staerd fylkis og haegri handar
A=zeros(M*N,M*N);         % Fylkid okkar er jafn stort og fjoldi punkta
RHS=zeros(M*N,1);         % Haegri hlid hefur somu staerd og fylkid en er vigur
ki = 1;                   % Dummy breyta fyrir visun í Rs
kj = 1;                   % Dummy breyta fyrir vísun í Qs
%Setjum gildi i fylki og haegri hlid
%Byrjum a því að finna nalgunargildi á R og Q
x_vigur = linspace(a,b,M);
y_vigur = linspace(c,d,N);
r_hnit=NaN;
p_hnit=NaN;
if ~isempty(R)
    r_hnit = zeros(length(R(1,:)),length(R(:,1)));
    for i = 1:length(R(:,1))
        r_hnit(i,1) = min(find(x_vigur>=R(i,1)));
        r_hnit(i,2) = min(find(y_vigur>=R(i,2)));
    end
```

```
end
if ~isempty(P)

    p_hnit = zeros(length(P(1,:)),length(P(:,1)));
    for j = 1:length(R(:,1))
        p_hnit(j,1) = min(find(x_vigur>=P(j,1)));
        p_hnit(j,2) = min(find(y_vigur>=P(j,2)));
    end
end



for n=1:N
    for m=1:M

        % Vid kjosum ad rada T_m,n thar sem fyrst er n=1 og m haekkar fra 1 til M,
        % Thvi naest haekkum vid n i 2 og haekkum m fra 1 til M, haekkum tha n i 3 og
        % haekkum m fra 1 til M og svo framvegis thar til m is jafnt M

        s=m+M*(n-1);                % Numer linu fylkis sem samsvarar m,n
        s_m_plus1=m+1+M*(n-1);      % Numer linu fylkis sem samsvarar m+1,n
        s_m_minus1=m-1+M*(n-1);     % Numer linu fylkis sem samsvarar m-1,n
        s_n_plus1=m+M*(n+1-1);      % Numer linu fylkis sem samsvarar m,n+1
        s_n_minus1=m+M*(n-1-1);     % Numer linu fylkis sem samsvarar m,n-1
        x = a+(m-1)*h;
        y = c+(n-1)*h;
        x_half = a+(m-1)*(h/2);
        y_half = c+(n-1)*(h/2);
        % Viðbót 2 - fast gildi í ákveðnum punkti

        % Jaðarskilyrði - u = gamma
        % Skilyrdi: (n==1) -> Viljum skoda nedstu linuna kassans
        % Skilyrdi: (n==N) -> Viljum skoda efstu linuna kassans
        % Skilyrdi: (m==1) -> Viljum skoda vinstri hlid kassans
        % Skilyrdi: (m==M) -> Viljum skoda haegri hlid kassans
        if  (m==1) || (M==m) || (n==1) || (N==n)
            A(s,s) = 1;
            RHS(s) = gamma(x,y);
            fprintf('Jadarpunktur n=%.0f , m = %.0f\n',n,m)
        end

        % Innri hnutpunktar kassans - jafna 22.3
        % Skilyrdi: (n>1)  -> Viljum ekki skoda nedstu linuna
        % Skilyrdi: (m>n) -> Viljum lenda haegra meginn vid vinstri
```

```
        %                    hlidarlinu thrihyrningsins
        % Skilyrdi: (m<M-n+1) -> Viljum lenda vinstra meginn vid haegri
        %                        hlidarlinu thrihyrningsins
        % Skilyrdi: (n<N) -> Viljum ekki lenda i topppunktinum.
        if (n>1) && (m>1) && (m<M) && (n<N)
            A(s,s) = ((p(x_half+h,y)+p(x_half-h,y)+p(x,y_half-h)+p(x,y_half+h))/h^2)+
            A(s,s_m_plus1) = -p(x_half+h,y)/h^2;   %p_j,r
            A(s,s_m_minus1) = -p(x_half-h,y)/h^2;  %p_j,l
            A(s,s_n_plus1) = -p(x,y_half+h)/h^2;   %p_j,t
            A(s,s_n_minus1) = -p(x,y_half-h)/h^2;  %p_j,s
            RHS(s) = f(x,y);
            fprintf('Innripunktur n=%.0f , m = %.0f\n',n,m)
            if ~isnan(p_hnit)
                if m == p_hnit(kj,1) && p_hnit(kj,2) == n && kj<=length(P(:,1))
                    RHS(s) = f(x,y)+sum(Q(kj)/h^2);
                    kj=kj+1;
                    fprintf('Sérstöðupunktur - Punktuppspretta x=%.2f , y = %.2f , n=
                end
            end
        end
        if  ki<=length(R(:,1))
            if m == r_hnit(ki,1) && r_hnit(ki,2) == n
                A(s,:) = 0;
                A(s,s) = 1;
                RHS(s) = U(ki);
                ki = ki+1;
                fprintf('Sérstöðupunktur - FAST GILDI x=%.2f , y = %.2f , n=%.2f , m
                disp(A(s,:))
            end
        end
    end
end
%Finnum lausn
A_sparse=sparse(A);
W_list=A_sparse\RHS;   % Thetta er vigur med hitastigum rodudum eins og vid til-
% greindum ad ofan
T_grid=zeros(N,M);
for n=1:N
    for m=1:M
        s=m+M*(n-1);   % Numer linu fylkis sem samsvarar m,n
        T_grid(n,m) = W_list(s); % Setjum inn gildi sem samsvarar
        % gildi fyrir thrihyrninginn okkar.
    end
end
```

```
W=T_grid;
```

3.2

Útskýrið hvers vegna uppsprettuliðirnnir eru meðhöndlaðir með því að setja $b_j = f(x_j, y_k) + h^{-2}$ ef (x_j, y_j) er sá punktur netsins sem næstur er P_s.

Lausn:

Við erum með net með uppsprettupunkti. En til þess að geta fundið hann þá þurfum við að vita fyrirframstaðsetingu á uppsprettupunktinum. Það gerum við með eftirfarandi. við vitum að

$$-\nabla(p\nabla u) + qu = f + Q\delta_{Ps}$$

ef uppsprettupunkturinn δ_{Ps} er í aðeins einum fyrirfram þekktum punkti þá vitum við að

$$\int\int \delta_{Ps}(x,y)dA = 1$$

svo ef við heildum yfir jaðarinn í kringum punktuppsprettuna þá fá um við út $..... + q_i * h^2 = f_i * h^2 + Q_s$ Síðan stöðlum við þessa jöfnu með því að deila öllu með h^2 og þá fáum við út

$$b_i = f_i + \frac{Q_s]}{h^2}$$

3.3

Þið þurfið að sannfæra ykkur um forritið reikni rétt. Tökum t.d. fallið $u(x,y) = x^2 - y^2$ veljið D=[-1,1]x[-1,1], og $\gamma(x,y) = x^2 - y^2$ á öllum jaðrinum, þá eigið þið að fá nálgunarfall sem er nánast eins og rétta lausnin.

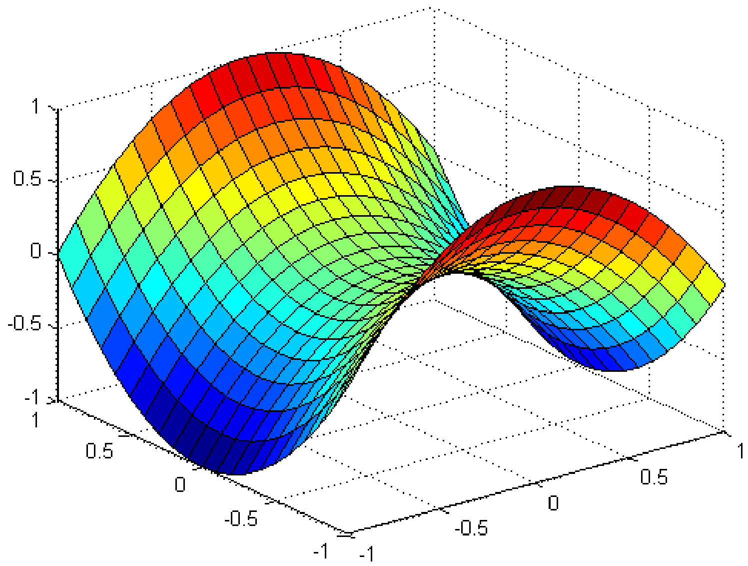
Lausn:

byrjum á því að skilgreina breyturnar okkar.

```
a=-1;
b=1;
c=-1;
d=1;
gamma= @(x,y) x.^2-y.^2;
p=@(x,y) 1;
q=@(x,y) 0;
f=@(x,y) 0;
h= 0.1;
```

Köllum svo á fallið:

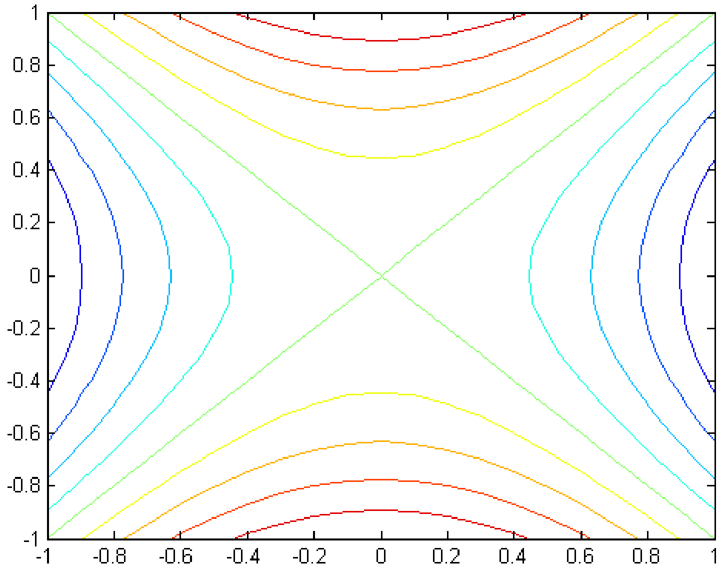
```
>> [W,A,RHS]=Dirichletverkefni(a,b,c,d,p,q,f,gamma,[1,1],[0],[1,0],[1],0.1)
>> surf(linspace(a,b,length(W(:,1))),linspace(c,d,length(W(1,:))),W');
```



Mynd 5: Söðulmynd

þessi liður gefur okkur svo contour af fallinu.

```
>> contour(linspace(a,b,length(W(:,1))),linspace(c,d,length(W(1,:))),W');
```



Mynd 6: Contour-mynd af Söðulmynd

3.4

Notið forritið ykkar til að þess að taka svæðið á D]-1,1]x[-1,1[, og reikna út og teikna upp lausinina á

$$\begin{cases} \nabla^2 u = 0 \\ \alpha = 1, \beta = 0, \gamma(x,y) \end{cases} = \begin{cases} 5max\left\{cos(\pi y), 0\right\}, x = \pm 1, \\ 5max\left\{cos(\pi y), 0\right\}, y = \pm 1. \end{cases}$$

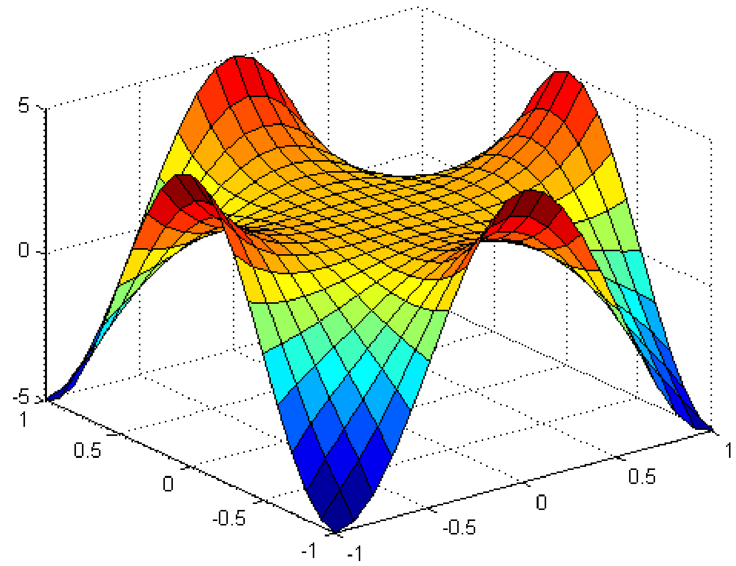
Setjið teikningu af grafi lausnarinnar í skýrsluna.
Lausn:
Byrjum á því að skilgreina breyturnar.

```
a=-1;
b=1;
c=-1;
d=1;
gamma= @(x,y) x.^2-y.^2;
p=@(x,y) 1;
q=@(x,y) 0;
f=@(x,y) 0;
h= 0.1;
```

Svo köllum við á fallið:

```
>>[W,A,RHS]=Dirichletverkefni(a,b,c,d,p,q,f,gamma,[],[],[],[],0.1)
>> surf(linspace(a,b,length(W(:,1))),linspace(c,d,length(W(1,:))),W');
```

Sjáum að myndin líkist kirkjunni í Kópavogi.

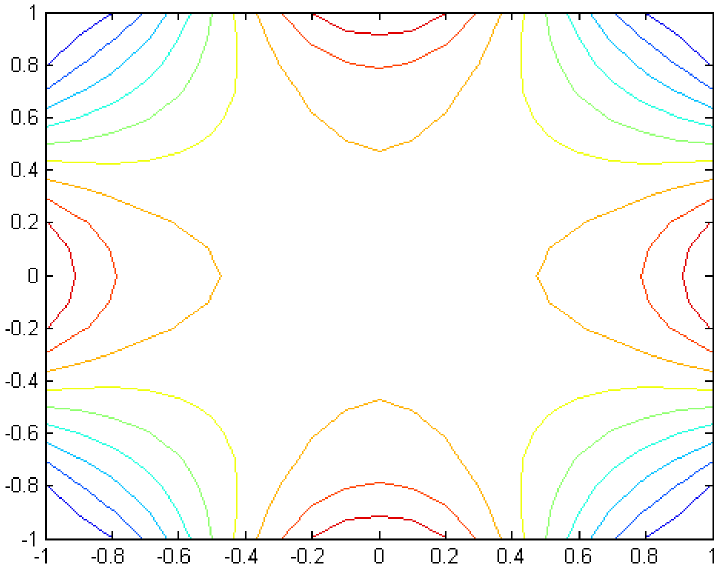


Mynd 7: Guðshúsið í Kópavogi

Köllum svo aftur á fallið til að fá contour-mynd.

```
>> contour(linspace(a,b,length(W(:,1))),linspace(c,d,length(W(1,:))),W');
```

Fáum þá út myndina.



Mynd 8: Countour-mynd af Guðshúsinu í Kópavogi

3.5

Líkið eftir Hróarskeldutjaldinu. Útskýrið hvaða formúlur þið notað og setjið myndí skýrsluna.

Lausn:

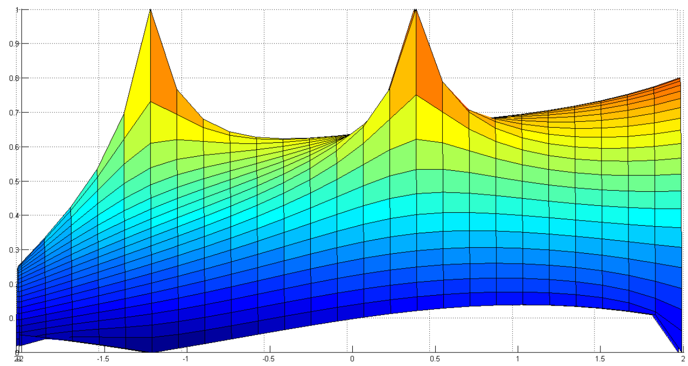
Byrjum á því að skilgreina eftirfarandi breytur:

```
a=-2;
b=2;
c=-0.5;
d=2;
gamma=@(x,y) hroarskelda_gamma(x,y);
p = @(x,y) 1+0.*x+0.*y;
q = @(x,y) 0+0.*x+0.*y;
f = @(x,y) 0+0.*x+0.*y;
h=0.1;
```

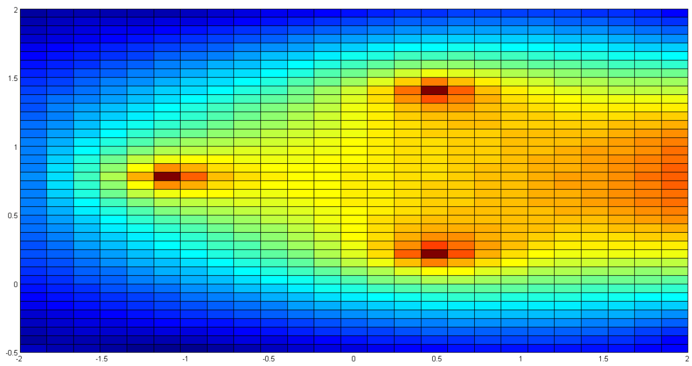
Aukaforritið `hroarskelda_gamma.m` er hægt að finna í viðauka hér að neðan. Við köllum svona á forritið til að fá eftirfarandi mynd:

```
>>[W,A,RHS]=Dirichletverkefni(a,b,c,d,p,q,f,gamma,[],[0],[0,0;-0.9,1;0.9,1],[1,1,1],
>> surf(linspace(a,b,length(W(:,1))),linspace(c,d,length(W(1,:))),W');
```

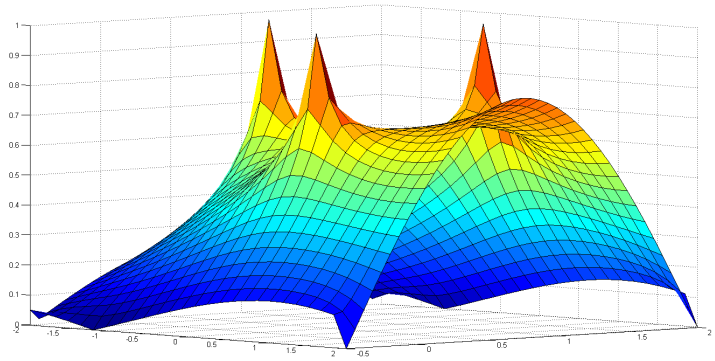
Sjáum hér að ofan hvað `a,b,c,d,p,q,f,gamma` er skilgreint sem.
Við völdum að þar sem punktuppspretturnar myndu byrja væru hnitin `[0,0],[-0.9,1],[0.9,1]` síðan hvað "magnitude-ið"væri á þessum punktuppsprettum en allar höfðu þær styrkinn 1. Út kom þá eftirfarandi myndir.



Mynd 9: Hróarskeldutjaldið á hlið



Mynd 10: Hróarskeldutjaldið ofan frá



Mynd 11: Hróarskeldutjaldið í allri sinni dýrð.

3.6

Hannið nýtt tjald til þess að setja upp við Arnarhól á menningarnótt í Reykjavík. Útskýrið hönnunina og setjið mynd í skýrsluna.
Lausn:

4 Lágmarksfletir

4.1

$(x, y) = \sqrt{\cosh(x)^2 - y^2}$ uppfyllir hlutafleiðujöfnuna fyrir lágmarksflöt. Notið það á rétthyrningnum D = [-1,1] * [-1,1] til þess að kanna hvort forritið reiknar.
Lausn:
Byrjum á því að skilgreina allar breytur.

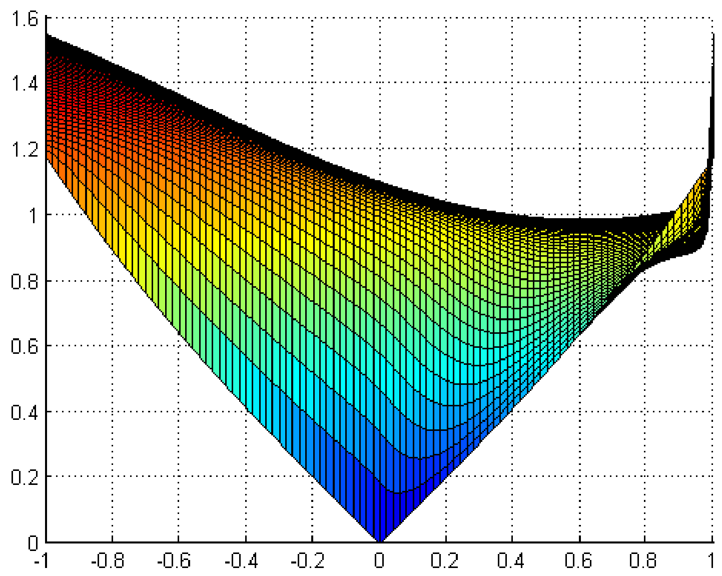
```
a=0;
b=1;
c=-1;
d=1;
gamma = @(x,y) sqrt(cosh(x).^2-y.^2);
p = @(x,y) -1+0.*x+0.*y;
q = @(x,y) 0+0.*x+0.*y;
f = @(x,y) 0+0.*x+0.*y;
[W_dummy,A,RSH] = Dirichletverkefni(a,b,c,d,p,q,f,gamma,[],[],[],[],0.02);
W0 = Dirichletverkefni(a,b,c,d,p,q,f,gamma,[],[],[],[],0.02);
k=1;
eps=0.01;
kmax = 5;
error = 2*eps;
```

Því næst keyrum við eftirfarandi forrit í matlab `lagmarksflotur.m` er hægt að finna í viðauka.

```
while(error > eps && k < kmax)
   W1=lagmarksflotur(a,b,c,d,p,q,f,gamma,[],[],[],[],0.02,W0');
   error = max(max(abs(W1-W0)));
   W0 = W1;
   k=k+1;
end
```

```
>> surf(linspace(a,b,length(W1(:,1))),linspace(c,d,length(W1(1,:))),W1');
```

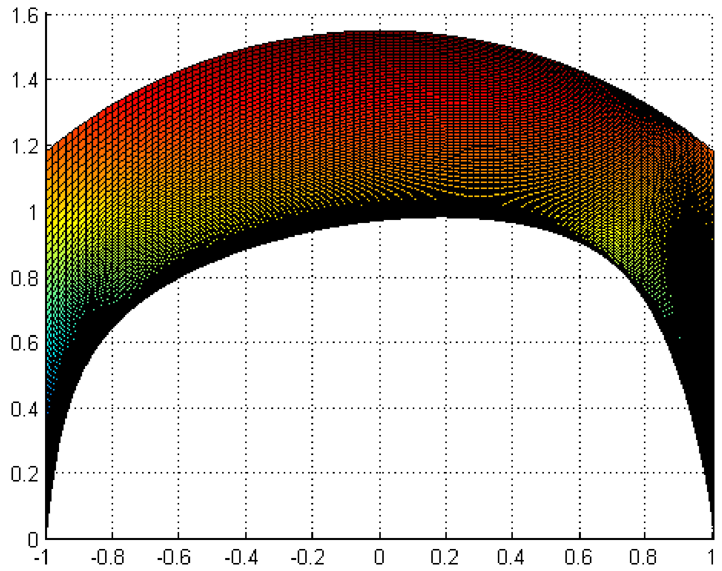
næst plottum við myndina.



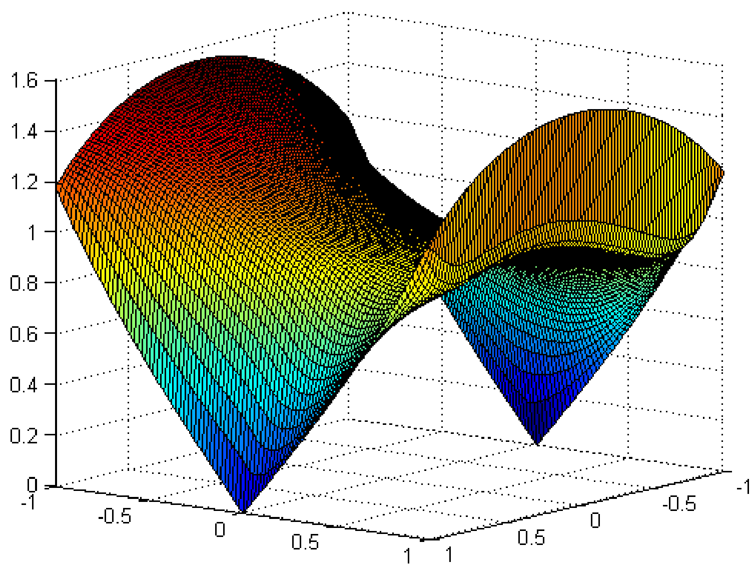
Mynd 12: Hliðarmynd af fletinum.

Köllum á contour-mynd af fallinu.

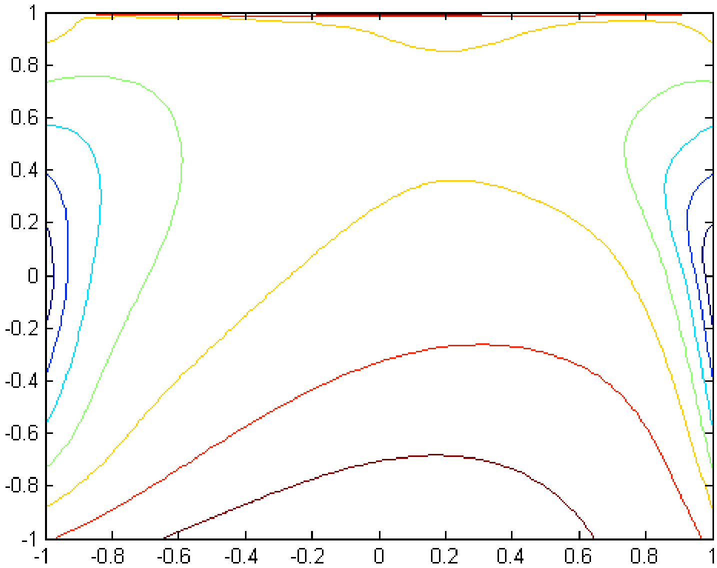
```
>> contour(linspace(a,b,length(W1(:,1))),linspace(c,d,length(W1(1,:))),W1');
```



Mynd 13: Framan á flötinn.



Mynd 14: Flöturinn



Mynd 15: Contour-mynd af fletinum.

4.2

Gerið grein fyrir stærð netsins sem þið notið, hvaða þolmörk þið notið til þess að stöðva ítrekunina og hversu langan tíma keyrslan tók á tölvunni ykkar.
Lausn:
Stærð netsins hjá okkur er W1=101x101. Þolmörkin sjáum við í eftirfarandi kóða

```
while(error > eps && k < kmax)
   W1=lagmarksflotur(a,b,c,d,p,q,f,gamma,[],[],[],[],0.02,W0');
   error = max(max(abs(W1-W0)));
   W0 = W1;
   k=k+1;
end
```

Sjáum að við keyrum þetta 5x í gegn. Notum `tic toc` skipunina til að finna út nákvæma tímasetningu og hún er 6.787692 sekúndur.

4.3

Berið saman á mynd hver munurinn er á því sem fékkst út í upphafsskrefinu og lokaskrefinu. Þetta má einfaldlega gera með því að teikna upp mismuninn.

4.4

Keyrið forritið með dæmið ykkar um Hróarskeldutjaldið eða Arnarhólstjaldið. Hver er munurinná lausninni á jöfnunni fyrir lágmarksflöt miðað við graf lausnarinnar á Laplace-jöfnu meðsömu jaðarskilyrðum. Setjið fallega mynd á forsíðuna.

12. apríl 2015